SIEMENS

RETURN THIS MANUAL

IF NOT RETURNED,
YOU WILL BE CHARGED
A MINIMUM OF $50.00

PLEASE RETURN TO:
AT&T CAPITAL CORPORATION
INSTRUMENT & DATA SERVICES
1830 WEST AIRFIELD DRIVE
DFW AIRPORT, TX 75261

14004-9 23 March 95

SIEMENS

Protocol Tester K1197®
Dual Interface

Basic Manual

Version Juli 1993

C73000–G6076–C121

© Siemens AG 1993

Customer Report **Protocol Tester K1197**

To

Siemens AG

ANL A425–KRK

Siemensalle 84

D–76187 Karlsruhe

TEL +49 (0) 7 21–5 95 44 75
Fax +49 (0) 7 21–5 95 66 30

Tx
Ttx

From

_____

_____

_____

_____

_____

TEL
Fax

Tx
Ttx

Suggestions, Reports, Improvements:

- - - -

Customer Report — Protocol Tester K1197

Soft– and Hardware used for test:

Application:_____ K1197 Type:_____

Version:_____ Serial–No.:_____

System+ Menu:_____

Options: ☐ Hard Disk ☐ RAM Extention ☐ other:_____

Tester Configuration:

☐ K1197 for monotoring

☐ K1197 for emulation/simulation as ☐ DCE / NT ☐ DTE / TE

Test environment: ☐ to another K1197

☐ other (Network, FAX Machine, etc.):_____

_____

_____

Error Description:

The error occurs: ☐ always ☐ sometimes

Encloses:

☐ Floppy Disk
☐ Testscript Listing
☐ Hardcopy/Printout
☐ other:

Date: **Signature:**

Contents

Protocol Tester K1197®
Dualinterface
D–Channel Application

User Manual

1	Release Notes	C73000–X6076–C121
2	ISDN Monitor	C73000–M6076–C174
3	ISDN Emulation/Simulation	C73000–M6076–C175
4	ISDN Testmanager	C73000–M6076–C176
5	Index	C73000–M6076–C195
6	ISDN Layer3 Variants	
7	X.25 D–Channel	C73000–M6076–C501
8	PHI Testpack	C73000–M6076–C523
9	CCS#7 – ISDN Interworking	C73000–M6076–C554
10	MSC – Simulator User Manual	

Copying of this document and giving it to others and the use or communication of the contents thereof,are forbidden without express authority. Offenders are liable to the payment of damages. All rights are reserved in the event of the grant of a patent or the registration of a utility model or design.

Subject to technical changes

Order No.: C73000–G6076–C121–9
Available from AUT3 K AZ Berlin
Printed in the Federal Republic of Germany
AG 0693 0.05 OI En

Dear Customer!

Please help us to improve our products. If you think, there's anything missing, or if you have any requests, please send the attached sheet to your local representative located nearest to your office and ask him to forward it to SIEMENS Karlsruhe, ANL A425–KRK, Germany as soon as possible.

For every error report we need the following information to find a solution as soon as possible:

- version number of the operating system and application software,
- detailed description of the error,
- list of commands typed in before the error occured,
- hardcopy of the screen when the error occured,
- hardcopy of the return stack backtrace, if it occured (press RETURN after command line window becomes red background),
- listing of the testscript, if the error was caused through in the testmanager,
- data recording file, if the error occured in the monitor.

Dear Reader !

We provide you with technical documentation to support our products.

With your cooperation we can improve our product information if you answer the following questions:

Which definitions or descriptions do you not understand?	
What supplementary or additional information would you like to see included?	
Have you found any mistakes?	
Have you found any printing errors?	

Please send your answer and other suggestions to:

Siemens AG
AUT3 MWB KQP
D–13623 Berlin

To
Local Siemens Sales Office

Repeat order for device:

MLFB No. 7KK1197–1 _ _ _ _

(Machine–readable product designation No.)

Serial No. BF _ _ _ _ – _ _

(see rating plate on rear of device)

The following products are ordered:

Item
1
2
3
4

Program:	ISDN Monitor / Simulation / Emulation
Version:	V 3.0
System:	MF 2.2
Previous Version:	V 2.0 α
Revision Date:	February 1991

☞ *Please note that it is important to read this paper before reading the manual. Some functions may*
☞ *be used in another way as described in the manual since we have had some changes after*
☞ *printing the manual.*

Problems fixed

1. The buffer problem to run two ISDN applications on two primary rate interfaces is now resolved. A memory extension mechanism (file ISDN_RAMX.F) is available to extend the dictionary for self defined definitions and testscripts.

2. A new timer mechanism in the operating system prevents the timeout event of a timer which is already stopped.

3. The command FILE->BUFFER now works properly.

4. The TRANSFER command with TRA_START and TRA_END now works correctly.

5. The TRACE output for traces longer than one line now works properly.

6. The timestamps of playback from multiport data recording files are now displayed correctly.

7. The decoder variables MADDR1*, MADDR2*, MCONTROL1* and MCONTROL2* now will be set correctly.

8. The B/D-channel communication now works properly.

9. The initialization of the impedance in simulation mode on primary interface is now always 120 ohm.

10. The problem of external B-channel access is resolved now.

11. Highspeed frame transmission without loss of frames is now possible.

Layer2:

1. The layer 2 emulator now works properly in states 7.0 - 7.7 and 8.0 - 8.7 in case of receiving I frames with N(R) error.

2. The layer 2 emulator now works properly in case of T203 timeout and no response of the peer data link layer entity. In this case the RR or RNR will be repeated three times (N200).

3. A DL-REL#CONF is now sent to the layer 3 (put into the L3 queue) in any cases of "local" confirmation (e.g. in case of DL_RELEASE in L2 state 4).

Layer3:

1. After loading a national variant for layer 3, the hash tables for QUICKY will be updated automatically.

2. The layer 3 multiplexor now works properly if a frame should be handled by more than one connection. The EVENT-TYPE will be preserved.

3. By generating a message using the message building system, the call reference value and protocol discriminator of the currently active connection will no longer be overwritten by the contents of the generated message.

4. The words =CR_LENGTH and =CN_CR_VALUE now work properly.

5. The use of the temporary buffers I#TEMP0 .. I#TEMP7 for message generation in automatic mode no longer causes an application crash. These buffers will be ignored by generation.

6. In the Q931 and ETSI variant, the IE "Channel id" is now mandatory in the SETUP message in the network-to-user direction according to the specification.

7. The decoder of 1TR6 IE "Network facilities" now works correctly.

8. In the VN2 variant, the octet numbers of the IE "Channel id" for primary rate interface are now correct.

Changes made

1. After loading a national variant for layer 3, all testmanagers will be stoped (GLOB_TM_STOP) and cleared (GLOB_TCLR).

Layer2:

1. For primary rate application and point-to-point data links of basic access, the ML-state machine is now off by default.

Layer3:

1. The layer 3 multiplexor is changed in case of incoming "broadcast" messages (TEI=127). In this case the CES is no longer a multiplex criteria and the testmanager will only be called once.

2. For 1TR6 and DKZE variants, the global call reference is no longer supported by the layer 3 multiplexor, because the global call reference is not defined by these specifications.

Enhancements

1. A new menu driven message building system is now available which allows the user easily to generate messages and information elements and modify, display and store these messages in the message pool.

2. The ISDN application software may now also run on V/X interface. A menu for the interface setup configuration is available.

3. The following national variants for layer 3 are now available and may be loaded by the menu system where the disk drive of the binary file is also selectable:

Q931	(CCITT Q.930, Q.931 BLUE BOOK)
ETSI	(ETS [CA] T/S 46-30)
1TR6	(FTZ 1TR3, Ausgabe 7.87, incl. Korrekturblätter)
DKZE	(FTZ 12TR7, Ausgabe 4.88, incl. Korrekturblätter)
CORNET-N	
CORNET-T	
VN2	(CNET spec. ST/LAA/ELR/TRA/733 & 743, 9.89)
AT&T	(5D5-900-321, 5ESS switch, 3.89)
	(PUB 41459, 4ESS switch, 4.86)
NTT	(INSNET, Japan)

4. Conformance test packets are available via menu and a user entry to the menu system is available for user defined menus for conformance tests or testscenarios.

5. X.25 in the D-channel is now available and all features are selectable via menu. X.25 is based on the CCITT specification of 1980 and 1984, a full emulation/simulation is possible on 8 logical channels per link.

6. A lot of HELP menus are available now.

7. Cursor key may now be handled by a testscript. The command MENU_OFF will suppress the opening of the menu system; the command MENU_ON will allow to open the menu system again.

8. Keyboard entries (string and function key events) which will by default be handled by the currently active testmanager may be handled only by a supervisor testmanager TM0. The command SVC_ON activates this function whereas the command SVC_OFF deactivates this function.

9. A mechanism is now available to generate an event within a testscript which can be recognized in another TM-state. The following words are available for this feature.

   SET_EVENT_FLAG (--)
   generates an event which will be recognized by ?TEST_EVENT.

   ?TEST_EVENT (-- flag)
   is the test manager recognition word which will return a true flag if an event has been generated by SET_EVENT_FLAG before.

   <TEST_EVENT_ACTION> (--)
   DOER word which will be executed by the occurence of the TEST event.

10. It is now possible to send frames with CRC error on basic access interface as well as on primary rate. By entering the command CRC_ERROR the following frame will be sent with CRC error. This feature is based on a patented, user specific hardware solution.

11. In between the words ACTION{ and }ACTION, STATE{ and }STATE, STATE_INIT{ and }STATE_INIT and SEQ{ and }SEQ, it will now be checked if an ENDIF of the IF - ENDIF construct is missing.

12. Some words are now available to change the contents of a message which is stored in the pool by commands.

EDIT_MSG	(name string -- flag) Takes the specified message from the pool and puts it into the coder area to allow to modify the message. A true flag is returned if no error has occurred.
STORE_MSG	(name string --) Writes the coder area into the pool entry specified by the name string. If the message did not exist before, the entry will be created.
GET_RX_MSG	(--) Takes the last received message and puts it into the coder area to allow to modify the message.
APPEND_IE	(IE-identifier --) Includes the specified IE into the message in the coder area.
DELETE_IE	(IE-identifier --) Removes the specified IE from the message in the coder area.

13. Some new decoder commands are now available for the testmanager on layer 3.

?L3_MAND_IE_MISSING (MSG-identifier -- t:f)
Searches missing mandatory IEs and if there are one or more IEs missing the word will return a true flag. The user can get the IEs by calling the word GET_IEs.

?L3_UNREC_IE (-- t:f)
Searches unrecognized IEs in the last decoded message. If there are one or more unrecognized IEs the word will return a true flag. GET_IEs will return these IEs.

?L3_MAND_IE_ERR (MSG-identifier -- t:f)
Searches for mandatory IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_OPT_IE_ERR (MSG-identifier -- t:f)
Searches for optional IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_MAND_IE_ORDER (MSG-identifier -- t:f)
Checks if all mandatory IEs ar in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

?L3_OPT_IE_ORDER (MSG-identifier -- t:f)
Checks if all optional IEs are in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

GET_IES (-- IE-code1/.../IE-code n/n | 0)
Returns the wrong IEs from the last called check word or a false flag if there was no error detected.

14. Some testscript examples are available as additional enhancements:
a.) Interworking testscenarios which can simulate A and B subscriber in both directions on one or two basic access interfaces. The name of the loader file for these scenarios is NT_L_TEL2.nat, where "nat" represents the corresponding layer 3 variant and may be Q931, ETSI, 1TR6, VN2, AT&T or X31D (X.25 in D-channel).
b.) Single scenarios which can simulate either A or B subscriber (which may be selected by function keys or by the loaded B-channel application). These scenarios also support the CODEC feature of the basic access interface for voice data. The name of the loader file for these scenarios is NT_L_TEL1.nat for all layer 3 variants (s.a.) and TE_L_TEL1.nat for Q931 and 1TR6 layer 3 variants.

ISDN MONITOR

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	January 1991

RELEASES NOTES R-1

1. INTRODUCTION 1-1

2. GENERAL 2-1

2.1. LOCAL CONTROL 2-1

2.2. REMOTE CONTROL 2-1

2.3. SET-UP 2-2

2.4. CHANGING DEFAULT START-UP CONFIGURATION 2-2

2.5. DISPLAY LEGEND 2-3

2.6. SPECIAL FUNCTIONS 2-4

2.7. SWITCHING FROM ONE APPLICATION TO ANOTHER 2-4

3. MONITOR STATES 3-1

3.1. HALT 3-1

3.2. PLAYBACK 3-2

3.3. FREEZE 3-3

3.4. MONITOR 3-4

4. MENU SYSTEM 4-1

4.1. MOVING AND SELECTING ITEMS 4-1

4.2. MENU ICONS 4-2

4.3. FUNCTION KEYS 4-2

4.4. MAIN MENU 4-3

4.5. REPORT SELECTION MENU 4-4

4.5.1. ERROR REPORT SELECTION MENU 4-5

4.6. TRIGGER / FILTER SELECTION MENU 4-6

4.6.1. FILTER SELECTION MENU 4-7

4.6.2. TRIGGER SELECTION MENU 4-8

4.7. DISK FUNCTION MENU 4-9

4.7.1. DATA RECORDING MENU 4-10

4.7.2. DATA PLAYBACK MENU 4-11

4.7.3. CAPTURE TO DISK MENU 4-12

4.8. SIMULATION MENU 4-12

4.9. TEST MANGER SELECTION MENU 4-13

4.10. PRINT MENU 4-14

4.11. SETUP MENU 4-15
4.11.1. CAPTURE RAM SETUP MENU 4-16
4.11.2. INTERFACE SETUP MENU (primary rate interface) 4-17
4.11.2.1. INTERFACE SETUP MENU (basic access interface) 4-18
4.11.2.2. INTERFACE SETUP MENU (V/X interface) 4-19
4.11.3. LAYER 3 PROTOCOL SETUP MENU 4-20
4.11.4. CONFORMANCE TESTS SETUP MENU 4-21
4.11.5. DISPLAY SETUP MENU 4-22
4.11.6. LOAD / SAVE SETUP MENU 4-23

4.12. UTILITIES MENU 4-24

4.13. STATUS MENU 4-24

4.14. CONFORMANCE TESTS MENU 4-24

5. REPORT GENERATOR 5-1

5.1. OVERVIEW 5-1

5.2. PHYSICAL LAYER REPORT MODES 5-2
5.2.1. DISPLAY OF COMMANDS AND INDICATIONS - C/I_ON (Basic access only) 5-2

5.3. FRAME LEVEL REPORT MODES 5-2
5.3.1. SHORT REPORT - FRM_SHRT 5-2
5.3.2. COMPLETE REPORT - FRM_COMP 5-2
5.3.3. HEX REPORT - FRM_HEX 5-2
5.3.4. CHARACTER REPORT - FRM_CHAR 5-2

5.4. MESSAGE LEVEL REPORT FORMATS 5-3
5.4.1. SHORT REPORT - MSG_SHRT 5-3
5.4.2. MEDIUM1 REPORT - MSG_MED1 5-3
5.4.3. MEDIUM2 REPORT - MSG_MED2 5-3
5.4.4. COMPLETE REPORT - MSG_COMP 5-3

5.5. MESSAGE DATA DISPLAY MODES 5-3
5.5.1. MNEMONIC DISPLAY - MSG_MNEM 5-3
5.5.2. HEX DISPLAY - MSG_HEX 5-3
5.5.3. CHAR DISPLAY - MSG_CHAR 5-3
5.5.4. NO DISPLAY - MSG_OFF 5-3

5.6. REPORT SCREEN EXAMPLES 5-4
5.6.1. Layer 1 Event Display Screens (Examples for Basic access) ... 5-4
5.6.2. Frame Event Display Screens 5-5
5.6.3. Message Event Display Screens 5-6

5.7. REPORT COMMANDS 5-9

5.8. RECEIVER OVERFLOW 5-10

6. DISK OPERATION 6-1
6.1. OVERVIEW 6-1
6.2. MULTIPORT DATA RECORDING 6-1
6.3. MULTPORT PLAYBACK SELECTION 6-2
6.4. PLAYBACK SELECTION COMMANDS 6-2
6.5. DISK COMMANDS 6-3

7. COMMANDS 7-1
7.1. GENERAL COMMANDS 7-1
7.2. TRANSFER COMMANDS 7-2
7.3. SCROLL CONTROL FOR CAPTURE AND DISK PLAYBACK 7-2
7.4. B-CHANNEL ACCESS 7-3
7.5. CONFIGURATION OF THE PRIMARAY RATE INTERFACE 7-3
7.6. CONFIGURATION OF THE V/X INTERFACE 7-4

8. PROGRAMMABLE FILTERS 8-1
8.1. REPORT FILTERS 8-1
8.1.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS) 8-1
8.1.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS) 8-2
8.1.3. FRAME LEVEL FILTER COMMANDS 8-3
8.1.4. MESSAGE LEVEL FILTER COMMANDS 8-4
8.1.4.1. PROTOCOL DISCRIMINATOR 8-4
8.1.4.2. CALL REFERENCE 8-4
8.2. CAPTURE FILTERS 8-5
8.2.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS) 8-5
8.2.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS) 8-6
8.2.3. FRAME LEVEL FILTER COMMANDS 8-7
8.2.4. MESSAGE LEVEL FILTER COMMANDS 8-8
8.2.4.1. PROTOCOL DISCRIMINATOR 8-8
8.2.4.2. CALL REFERENCE 8-8
8.3. DISK FILTERS 8-9
8.3.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS) 8-9
8.3.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS) 8-10
8.3.3. FRAME LEVEL FILTER COMMANDS 8-11

8.3.4. MESSAGE LEVEL FILTER COMMANDS 8-12
8.3.4.1. PROTOCOL DISCRIMINATOR 8-12
8.3.4.2. CALL REFERENCE 8-12

9. PROGRAMMABLE TRIGGERS 9-1

9.1. EVENT TRIGGERS 9-1

9.2. COMMANDS TO SPECIFY TRIGGER EVENTS 9-2
9.2.1. TRIGGER MACHINE COMMANDS 9-2
9.2.2. TRIGGER EVENT COMMANDS 9-2
9.2.2.1. TRIGGER DIRECTIONS 9-2
9.2.2.1.1. PHYSICAL LEVEL EVENTS (BASIC ACCESS) 9-2
9.2.2.1.2. PHYSICAL LEVEL EVENTS (PRIMARY RATE ACCESS) 9-3
9.2.2.2. FRAME LEVEL EVENTS 9-3
9.2.2.3. MESSAGE LEVEL EVENTS 9-4
9.2.2.4. MISCELLANEOUS EVENTS 9-4
9.2.3. EXAMPLES 9-5

9.3. DEFINING TRIGGER ACTIONS 9-6

INDEX I-1

Program:	ISDN Monitor / Simulation / Emulation
Version:	V 3.0
System:	MF 2.2
Previous Version:	V 2.0 α
Revision Date:	February 1991

☞ *Please note that it is important to read this paper before reading the manual. Some functions may*
☞ *be used in another way as described in the manual since we have had some changes after*
☞ *printing the manual.*

Problems fixed

1. The buffer problem to run two ISDN applications on two primary rate interfaces is now resolved. A memory extension mechanism (file ISDN_RAMX.F) is available to extend the dictionary for self defined definitions and testscripts.

2. A new timer mechanism in the operating system prevents the timeout event of a timer which is already stopped.

3. The command FILE->BUFFER now works properly.

4. The TRANSFER command with TRA_START and TRA_END now works correctly.

5. The TRACE output for traces longer than one line now works properly.

6. The timestamps of playback from multiport data recording files are now displayed correctly.

7. The decoder variables MADDR1*, MADDR2*, MCONTROL1* and MCONTROL2* now will be set correctly.

8. The B/D-channel communication now works properly.

9. The initialization of the impedance in simulation mode on primary interface is now always 120 ohm.

10. The problem of external B-channel access is resolved now.

11. Highspeed frame transmission without loss of frames is now possible.

Layer2:

1. The layer 2 emulator now works properly in states 7.0 – 7.7 and 8.0 – 8.7 in case of receiving I frames with N(R) error.

2. The layer 2 emulator now works properly in case of T203 timeout and no response of the peer data link layer entity. In this case the RR or RNR will be repeated three times (N200).

3. A DL-REL#CONF is now sent to the layer 3 (put into the L3 queue) in any cases of "local" confirmation (e.g. in case of DL_RELEASE in L2 state 4).

Layer3:

1. After loading a national variant for layer 3, the hash tables for QUICKY will be updated automatically.

2. The layer 3 multiplexor now works properly if a frame should be handled by more than one connection. The EVENT-TYPE will be preserved.

3. By generating a message using the message building system, the call reference value and protocol discriminator of the currently active connection will no longer be overwritten by the contents of the generated message.

4. The words =CR_LENGTH and =CN_CR_VALUE now work properly.

5. The use of the temporary buffers I#TEMP0 .. I#TEMP7 for message generation in automatic mode no longer causes an application crash. These buffers will be ignored by generation.

6. In the Q931 and ETSI variant, the IE "Channel id" is now mandatory in the SETUP message in the network-to-user direction according to the specification.

7. The decoder of 1TR6 IE "Network facilities" now works correctly.

8. In the VN2 variant, the octet numbers of the IE "Channel id" for primary rate interface are now correct.

Changes made

1. After loading a national variant for layer 3, all testmanagers will be stoped (GLOB_TM_STOP) and cleared (GLOB_TCLR).

Layer2:

1. For primary rate application and point-to-point data links of basic access, the ML-state machine is now off by default.

Layer3:

1. The layer 3 multiplexor is changed in case of incoming "broadcast" messages (TEI=127). In this case the CES is no longer a multiplex criteria and the testmanager will only be called once.

2. For 1TR6 and DKZE variants, the global call reference is no longer supported by the layer 3 multiplexor, because the global call reference is not defined by these specifications.

Enhancements

1. A new menu driven message building system is now available which allows the user easily to generate messages and information elements and modify, display and store these messages in the message pool.

2. The ISDN application software may now also run on V/X interface. A menu for the interface setup configuration is available.

3. The following national variants for layer 3 are now available and may be loaded by the menu system where the disk drive of the binary file is also selectable:

Q931	(CCITT Q.930, Q.931 BLUE BOOK)
ETSI	(ETS [CA] T/S 46-30)
1TR6	(FTZ 1TR3, Ausgabe 7.87, incl. Korrekturblätter)
DKZE	(FTZ 12TR7, Ausgabe 4.88, incl. Korrekturblätter)
CORNET-N	
CORNET-T	
VN2	(CNET spec. ST/LAA/ELR/TRA/733 & 743, 9.89)
AT&T	(5D5-900-321, 5ESS switch, 3.89)
	(PUB 41459, 4ESS switch, 4.86)
NTT	(INSNET, Japan)

4. Conformance test packets are available via menu and a user entry to the menu system is available for user defined menus for conformance tests or testscenarios.

5. X.25 in the D-channel is now available and all features are selectable via menu. X.25 is based on the CCITT specification of 1980 and 1984, a full emulation/simulation is possible on 8 logical channels per link.

6. A lot of HELP menus are available now.

7. Cursor key may now be handled by a testscript. The command MENU_OFF will suppress the opening of the menu system; the command MENU_ON will allow to open the menu system again.

8. Keyboard entries (string and function key events) which will by default be handled by the currently active testmanager may be handled only by a supervisor testmanager TM0. The command SVC_ON activates this function whereas the command SVC_OFF deactivates this function.

9. A mechanism is now available to generate an event within a testscript which can be recognized in another TM-state. The following words are available for this feature.

   SET_EVENT_FLAG (--)
   generates an event which will be recognized by ?TEST_EVENT.

   ?TEST_EVENT (-- flag)
   is the test manager recognition word which will return a true flag if an event has been generated by SET_EVENT_FLAG before.

   <TEST_EVENT_ACTION> (--)
   DOER word which will be executed by the occurence of the TEST event.

10. It is now possible to send frames with CRC error on basic access interface as well as on primary rate. By entering the command CRC_ERROR the following frame will be sent with CRC error. This feature is based on a patented, user specific hardware solution.

11. In between the words ACTION{ and }ACTION, STATE{ and }STATE, STATE_INIT{ and }STATE_INIT and SEQ{ and }SEQ, it will now be checked if an ENDIF of the IF - ENDIF construct is missing.

12. Some words are now available to change the contents of a message which is stored in the pool by commands.

EDIT_MSG (name string -- flag)
Takes the specified message from the pool and puts it into the coder area to allow to modify the message. A true flag is returned if no error has occurred.

STORE_MSG (name string --)
Writes the coder area into the pool entry specified by the name string. If the message did not exist before, the entry will be created.

GET_RX_MSG (--)
Takes the last received message and puts it into the coder area to allow to modify the message.

APPEND_IE (IE-identifier --)
Includes the specified IE into the message in the coder area.

DELETE_IE (IE-identifier --)
Removes the specified IE from the message in the coder area.

13. Some new decoder commands are now available for the testmanager on layer 3.

?L3_MAND_IE_MISSING (MSG-identifier -- t:f)
Searches missing mandatory IEs and if there are one or more IEs missing the word will return a true flag. The user can get the IEs by calling the word GET_IEs.

?L3_UNREC_IE (-- t:f)
Searches unrecognized IEs in the last decoded message. If there are one or more unrecognized IEs the word will return a true flag. GET_IEs will return these IEs.

?L3_MAND_IE_ERR (MSG-identifier -- t:f)
Searches for mandatory IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_OPT_IE_ERR (MSG-identifier -- t:f)
Searches for optional IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_MAND_IE_ORDER (MSG-identifier -- t:f)
Checks if all mandatory IEs ar in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

?L3_OPT_IE_ORDER (MSG-identifier -- t:f)
Checks if all optional IEs are in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

GET_IES (-- IE-code1/.../IE-code n/n | 0)
Returns the wrong IEs from the last called check word or a false flag if there was no error detected.

14. Some testscript examples are available as additional enhancements:

a.) Interworking testscenarios which can simulate A and B subscriber in both directions on one or two basic access interfaces. The name of the loader file for these scenarios is NT_L_TEL2.nat, where "nat" represents the corresponding layer 3 variant and may be Q931, ETSI, 1TR6, VN2, AT&T or X31D (X.25 in D-channel).

b.) Single scenarios which can simulate either A or B subscriber (which may be selected by function keys or by the loaded B-channel application). These scenarios also support the CODEC feature of the basic access interface for voice data. The name of the loader file for these scenarios is NT_L_TEL1.nat for all layer 3 variants (s.a.) and TE_L_TEL1.nat for Q931 and 1TR6 layer 3 variants.

1. INTRODUCTION

Since all of the ISDN Software packages contain an Monitor part in any configuration, this guide is intended to be the basic manual for all the configurations of the ISDN Software. It provides all necessary information for the operation of the ISDN Layer 2 and Layer 3 Monitor Software. The basic concepts are the same for both layers.

There are specific versions (according to a selected national variant) of the Layer 3 Monitor. See the LAYER 3 (INTER)NATIONAL VARIANT Manual part for the specific version your Layer 3 Monitor is set up.

This manual is independent from the selected interface. Most of the information is valid for Basic Access as well as Primary Rate Access and V/X interface. All differences in the Software between the interfaces is especially marked.

The ISDN Monitor provides for the comprehensive monitoring and display of ISDN D-channel protocol information at Layer 2 and Layer 3. In addition it is possible to monitor or record selected events by the use of triggers and filters.

The Monitor is completely **passive** only in the **MONITOR** configurations of the ISDN Software packages. Consequently it does not affect the signalling information that it monitors.

For **EMULATION** or **SIMULATION** configurations all Monitor functions are included, but the emulation will be still active.

Simultaneous display of the signalling information for both directions of a link is possible with the information from each direction identified by a different color and a label. The information received may be monitored (displayed), stored in a Capture RAM and/or recorded on Disk for analysis or use at later time.

The Monitor comprises several parts: the Monitor, Triggers and Filters. The Monitor provides for the display of ISDN Layer 2 and Layer 3 information, with a range of display formats at both Layers. These monitor formats operate independently of each other so that the user can have any combination of display formats desired. In addition to the wide range of display formats, there is a range of display modes, such as ASCII, HEX, Double HEX, etc.

Trigger and Filter functions are available such that it is possible to report, capture and record only those events which are of interest. Triggers can be set to begin, modify or end a monitoring session while Filters select only the required information to be displayed and/or stored.

The information saved in the Capture Ram can be scrolled to display and transferred to a disk if desired.

The information from the Disk recording can be played back, displayed or printed out.

All the features of the ISDN Monitor are accessible from within the Test Manager. Consequently it is possible to write FORTH code which collects statistics and analyzes protocol procedures while monitoring live data or playing back captured or recorded data.

Section 2 of this document describes the SET UP and general features for the ISDN Software packages. These information is related to all configurations of the ISDN Software.

Section 3 gives an overview of the different Monitor states (HALT, FREEZE, MONITOR), whereas Section 4 describes the various menus and the associated function keys.

Sections 5, 6, 7, 8 and 9 contain more detailed information on the operation of the Report generator, Disk operations, Filter and Trigger machines and the related commands that can be used for Test Manager scenarios or user Applications.

2. GENERAL

This section pertains to the MONITOR as well as to the EMULATION or SIMULATION versions of the ISDN Software package.

2.1. LOCAL CONTROL

Local control of K1197 applications can be done using the keyboard. This keyboard provides a full ASCII keypad, 8 function keys, and four cursor keys.

The ASCII entries are processed when the return key is entered. Entry will be processed by the FORTH interpreter to determine what commands are to be executed.

Function key entry is prompted by the function menus displayed on the bottom line of the screen. The line is the direct function key entry.

The cursor keys are used to scroll through the data when playing back from disk or capture ram. They are also used for entry selection when screen menus are being displayed.

2.2. REMOTE CONTROL

Remote control of K1197 applications can be done using the MODEM connector on the auxiliary interface module on the back panel of the K1197.

The remote control line is configured to use async, 8 bits, no parity, and 1200 baud. The K1197 does not echo received characters. If local display is wanted on a terminal, the terminal should be set to half duplex.

Commands on the remote control line are to be entered, followed by a carriage return character. The program will process the input, and send the command line display to the remote site, followed by the prompt " ROK " and the carriage return and linefeed characters. The programs will process he remote commands the same way as if they had been entered on the local keyboard.

Refer to the remote control section of the K1197 USER MANUAL for more information on special remote control commands.

2.3. SET-UP

Step 1 Make sure that the appropriate interface connector is attached for the hardware in the K1197.

Step 2 Make sure that the K1197 is connected to the equipment under test using the cables and connectors (e.g. eight pin Telephone Modular Connectors for Basic Access, for Primary Rate Access or V/X acces).

Step 3 Insert the K1197 System Disk into drive 0 for Protocol Testers without a hard disk and follow the directions for "SET UP" of the K1197 User Manual.

Step 4 Wait until the K1197 Power-up Menu appears on the screen. This shows "SIEMENS" in very large characters on the screen.

Step 5 Place the ISDN Test Pack diskette in either drive and from the Power-up Menu select the appropriate ISDN application (ISDN MONITOR).

The ISDN Monitor software has now been loaded.

Note: Setup of Emulation/Simulation versions is the same way except SIMULATION is used at step 5. The first entry to the Monitor/Emulation is done by selecting "ENTER APPLICATION" in the main menu. This will initialize all internal information. Subsequent entries into the monitor/emulator do not reinitialize this information.

2.4. CHANGING DEFAULT START-UP CONFIGURATION

The user can create a special ISDN setup file which will start-up the Application in a frequently used configuration. To do this, the user can set the ISDN Application in the required configuration and save this configuration in a file " ISDN_SETUP.x" (where x is the number of the corresponding ISDN application binary). This file will be loaded by entering the application. (See SETUP menu of the ISDN application in chapter 4.11.5 of this manual for more information.)

2.5. DISPLAY LEGEND

The general screen layout shown in figure 2 is applicable to all ISDN Applications.

Note that the ISDN Application screen layout is based on the Protocol Tester screen layout as shown in the K1197 User Manual. The special functions defined there are also applicable to the ISDN Application.

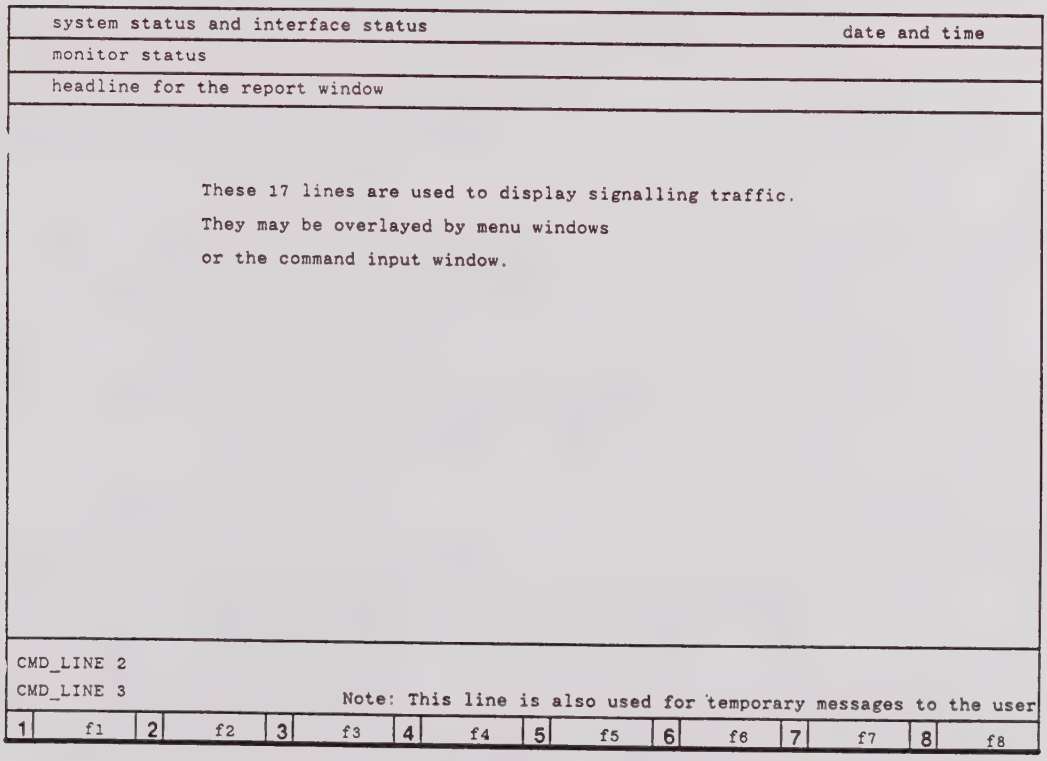

system status and interface status
date and time
monitor status
headline for the report window
These 17 lines are used to display signalling traffic.
They may be overlayed by menu windows
or the command input window.
CMD_LINE 2
CMD_LINE 3
Note: This line is also used for temporary messages to the user
1 f1 2 f2 3 f3 4 f4 5 f5 6 f6 7 f7 8 f8

Figure 2-1: K1197 – Screen layout

Line 1-3	Status lines – provide setup and configuration information
Line 4-20	General Data and menu window display area
Line 4-22	Command input lines (by default only the last two lines are visible)
Line 22	Message line
Line 23	**f1 – f8** : function keys

The first three lines are Status lines which provide setup and configuration information for the user. Data and menus can be displayed in the next seventeen lines. The next two lines are the last two lines of the command window. If the user chooses to enter Forth commands instead of, or in addition to, using the function keys, the keyboard entry and the system responses appear in this window. The final line is used to display the current function key assignments.

2.6. SPECIAL FUNCTIONS

This 3-key combination returns the system to the FORTH interpreter in case of a lock-up or program error.

This 3-key combination prints a hardcopy of the current screen display. The default printer configuration is 1200 baud, 8 bits/character, ASCII, DTR and XON/OFF control, LR CR end of line, 80 characters/line, 60 lines/page. See the printer section of the K1197 USER MANUAL for detailed information.

2.7. SWITCHING FROM ONE APPLICATION TO ANOTHER

Switching from one application to another, that is to change the display and key board control to another application program, can be done by pressing the key **ESC**. It toggles the control between application programs and the main utility functions (wrap around).

3. MONITOR STATES

The ISDN Monitor is always in one of three states:

1. HALT *(playback mode, data retrieved from Capture RAM or Diskette),*
2. FREEZE *(data retrieved from Capture RAM, direct data may be recorded to Diskette),*
3. MONITOR *(data arrives directly from the interface).*

On initial power-up, the Monitor is in the HALT state. These states can be changed by the f5, f6, f7 function keys or by the commands HALT, FREEZE, or MONITOR and PLAYBACK.

3.1. HALT

All processing of live data is halted. The contents of the capture RAM is displayed on the screen. Any unwanted information may be suppressed by using the report filters. Event triggers can also be active.

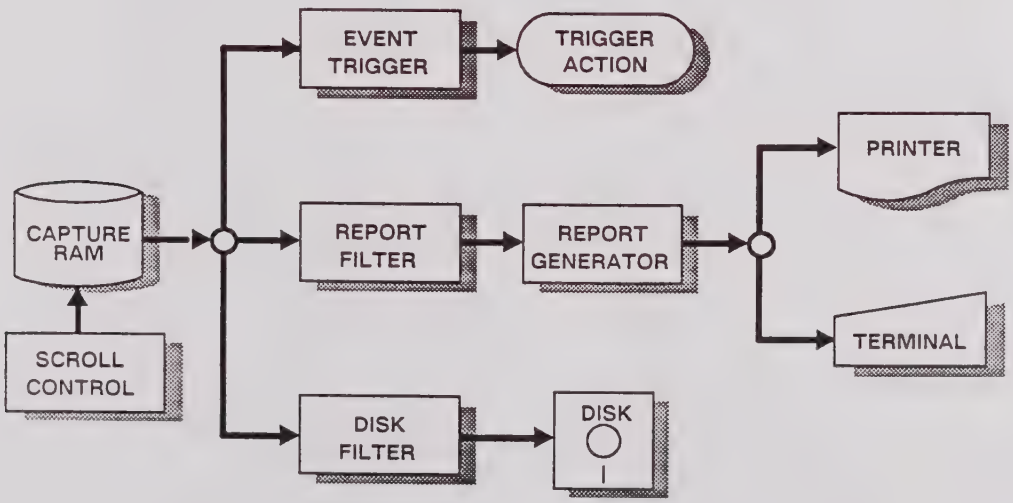


Figure 3-1: Data from Capture Ram in HALT mode

3.2. PLAYBACK

Live data is not processed. The contents of the selected data recording file is reviewed on the screen. The report filters and the playback selection are used to suppress unwanted information. Event triggers can be active.

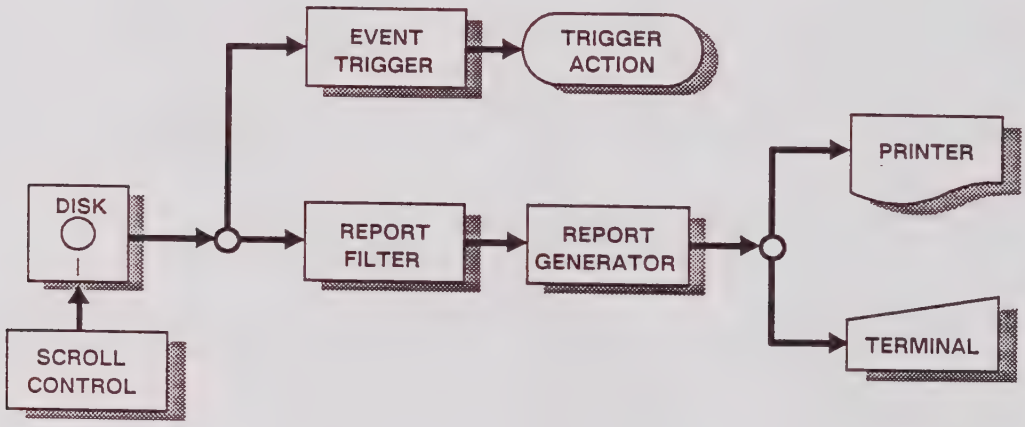


Figure 3-2: Data from Disk Playback in HALT mode

3.3. FREEZE

Live data is not written to capture RAM or sent to the report generator. However, live data can still be recorded on disk. The contents of the capture RAM can be reviewed on the display. Event triggers on the live data can be active.

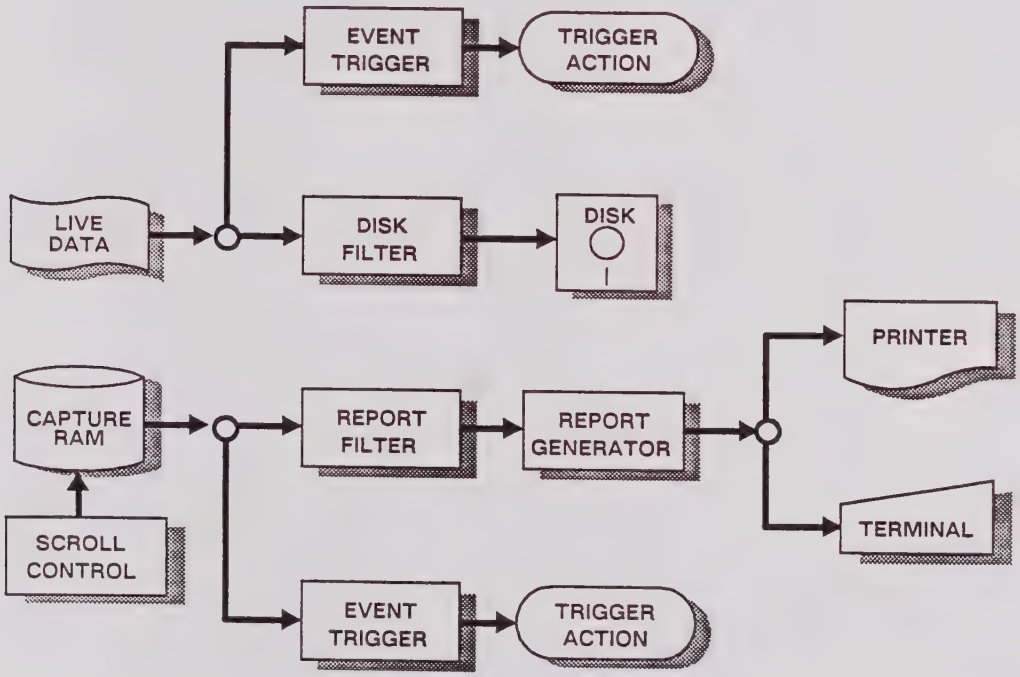


Figure 3-3: Data Flow in the Freeze Mode

3.4. MONITOR

Live data is displayed on the screen and can optionally be recorded to disk and/or stored in capture RAM.

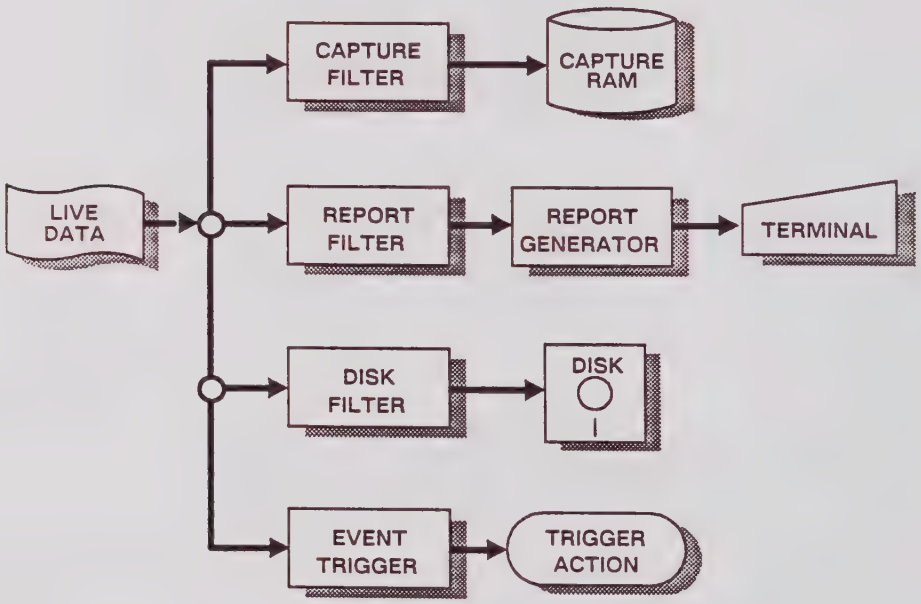


Figure 3-4: Data Flow in the Monitor Mode

4. MENU SYSTEM

The K1197 menu system provides an intuitive interface designed to minimize the need to refer to the manual. It includes pop-up help screens to remind the user of meanings of selections available in the menus.

4.1. MOVING AND SELECTING ITEMS

The menu system is activated by typing the MENU function key or by pressing the right arrow key on the keyboard. This brings up the first menu from which all the other menus may be accessed.

Within a menu, the following keys may be used to move and perform actions:

Cursor up (↑)	Move to next selectable item above this one.
Cursor down (↓)	Move to next selectable item below this one.
Cursor left (←)	Move to previous menu to the left without entering current item. Note: Some menus do not allow this key in order to force the user to select some action through a function key before leaving the menu (e.g: the Hex editor and Message Building System).
Cursor right (→)	Move to next menu to the right if one exists.
Control cursor up (ctrl-↑)	Move up one page on scrolling menus.
Control cursor down (ctrl-↓)	Move down one page on scrolling menus.
Control shift cursor up (ctrl-shift-↑)	Move to first item in menu.
Control shift cursor down (ctrl-shift-↓)	Move to last item in menu.
Control shift cursor left (ctrl-shift-←)	Close all opened menu windows except the first one.
Control shift cursor right (ctrl-shift-→)	Open all those menu windows which were closed by ctrl-shift-← or a series of cursor left actions.
Shift cursor right (shift-→)	In string and numeric entry fields: Shift the characters from this place on one step forward and create an empty place (blank) between the already existing characters.
Shift cursor left (shift-←)	In string and numeric entry fields: Delete the character under the cursor and shift the following characters one step backward.
Control cursor right (ctrl-→)	In string and numeric entry fields: Move the blinking cursor to the last character in the string.
Control cursor left (ctrl-←)	In string and numeric entry fields: Move the blinking cursor to the first character in the string.
Home	Move to first item in menu (same as ctrl-shift-↑).
Return	Select an item or complete entry of a typed value.
Back Space (BS), Delete (DEL)	In string and numeric entry fields, will erase the character before the cursor.
Space bar	When the cursor is on a field which takes a number of options, the space bar may be used to scroll through the available options. These options may also be accessed by typing cursor right and selecting the desired option by pressing RETURN.

4.2. MENU ICONS

The menus use a number of icons to convey information about the options available on the menu. The borders of the menu are of two types: single line and double line. Double line borders are used to indicate the last menu in a chain. There will not be any menus to the right of a double bordered menu.

At the top and bottom of the right border of a menu there will apear arrows if the menu is bigger than what will fit in the window. The arrow indicates that there are more items outside the window in the direction indicated.

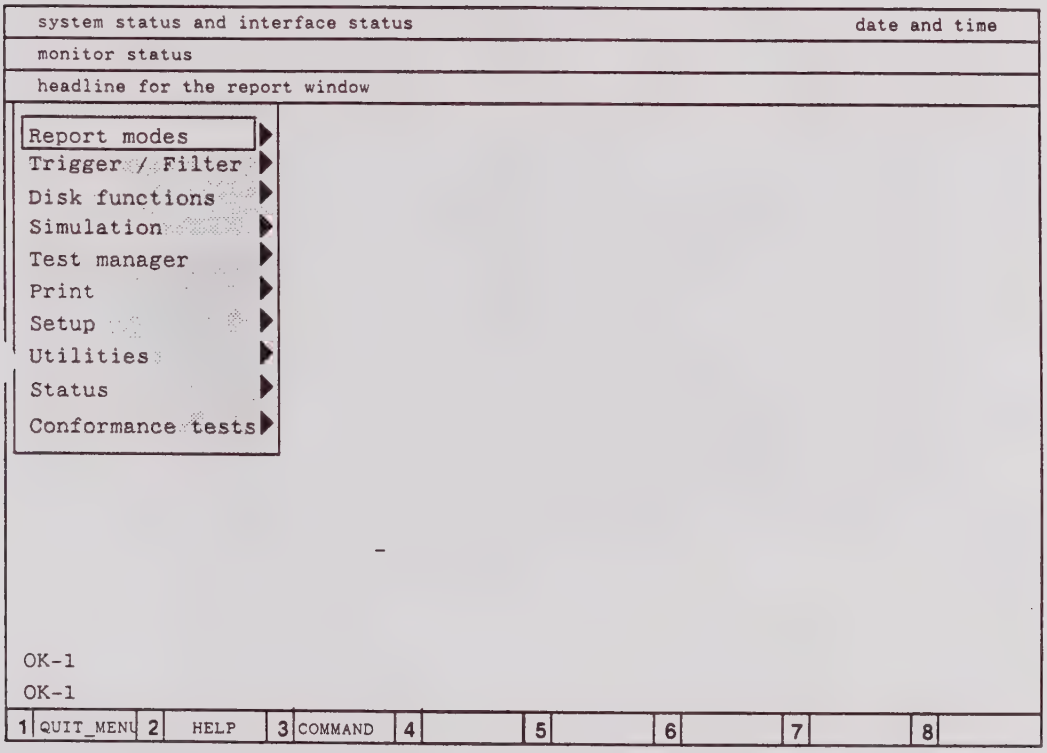
At the right of a selectable item, one of the following may appear: ▶ or ≫. The single solid arrow indicates that there is a menu to the right of this item that will be displayed when cursor right is typed. The double arrow indicates that the field takes a number of options which may be scrolled through with the space bar or listed using the cursor right key.

4.3. FUNCTION KEYS

A set of function keys on the bottom line of the display is almost always available. Some common keys are listed here.

MENU / QUIT_MENU	: Enter to menu system control / Leaves the menu system control
HELP / QUIT_HELP	: Open a help menu for the particular menu window.
COMMAND / QUIT_COM	: Give the control to the FORTH interpreter which enables the use of keyboard for typing commands. The menu system window remains still visible. Return the control back to the menu system.
HALT	: Switch to HALT mode
FREEZE	: Switch to FREEZE mode
MONITOR	: Switch to MONITOR mode
TM KEYS / MON KEYS	: Switch to Test manager control keys / back to monitor control keys

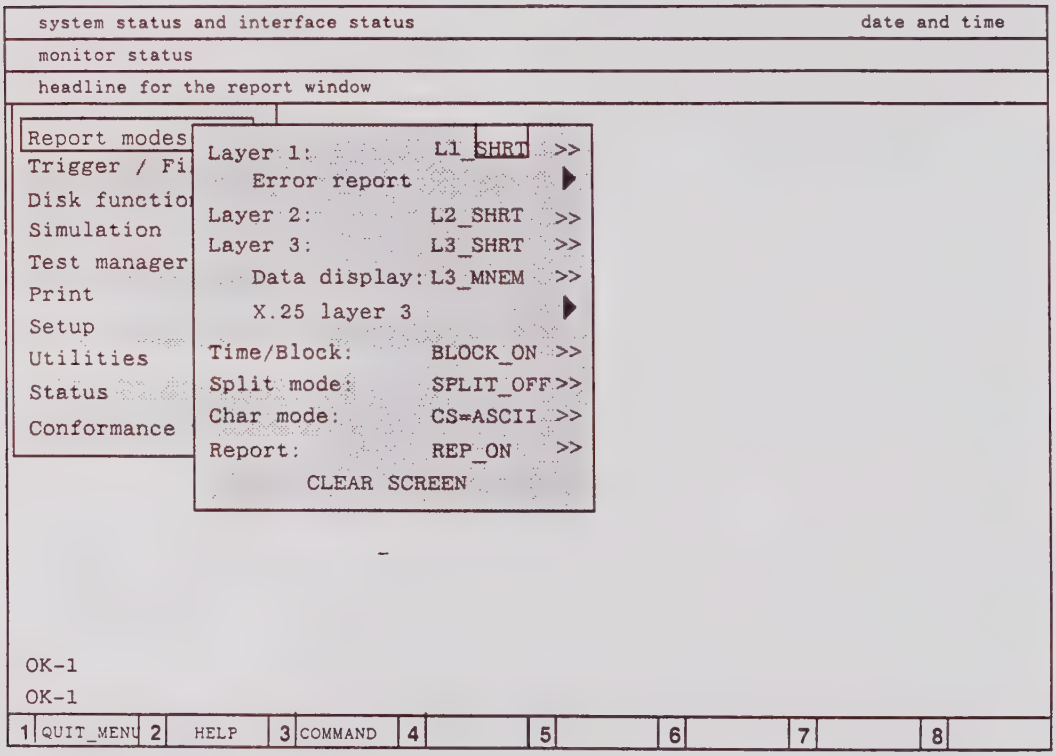
4.4. MAIN MENU



MENU SELECTION:

Report modes	: Enter report selection menu
Trigger / Filter	: Enter trigger and filter selection menu
Disk functions	: Enter disk function menu
Simulation	: Enter simulation menu (only available in simulator software)
Test manager	: Enter test manager selection menu
Print	: Enter printout selection menu
Setup	: Enter setup selection menu
Utilities	: Enter general utility menu
Status	: Enter status selection menu (only available in simulator software)
Conformance Tests	: Enter menu to select and start conformance tests (only available if conformance test software is loaded)

4.5. REPORT SELECTION MENU

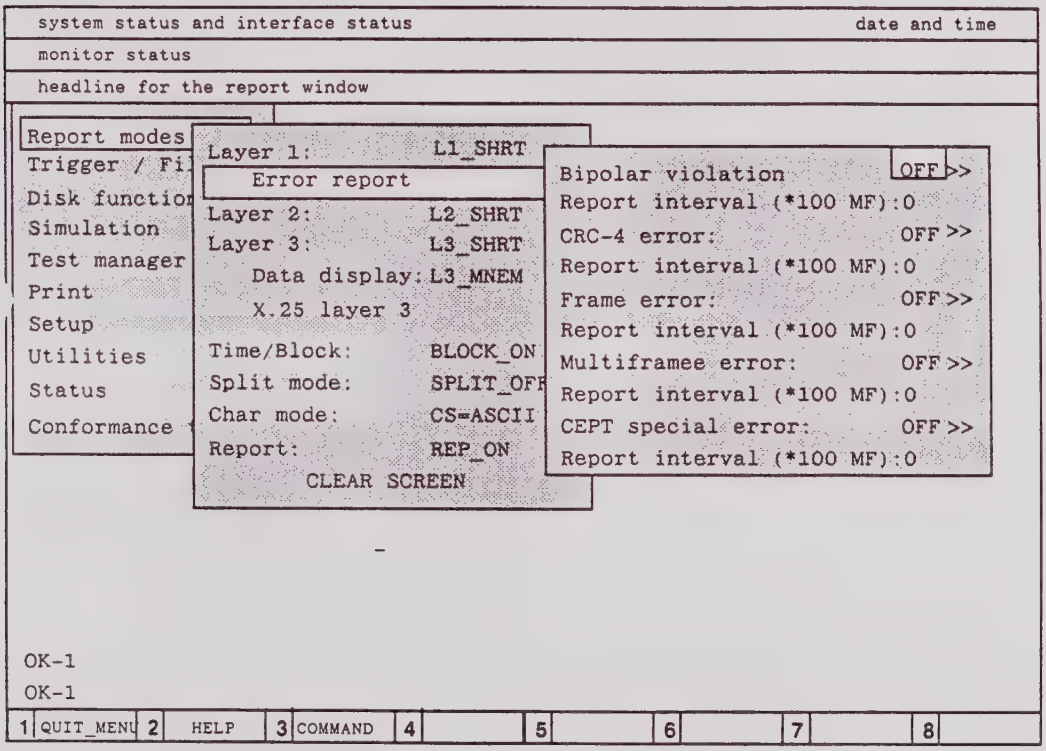


MENU SELECTION:

Layer 1	: Select layer 1 report format
Error report	: Enter menu for layer 1 error report selection (PRI only)
C/I-Monitor	: Select monitoring of layer 1 commands and indications (BAI only)
Layer 2	: Select frame report format
Layer 3	: Select message report format
Data display	: Select message data display format
X.25 layer 3	: Select display format of X.25 data in layer 3
Time/Block	: Select display of block number or timestamp
Split mode	: Select splitted screen mode
Char mode	: Select character display mode
Report	: Select report display
CLEAR SCREEN	: Clear data display screen

4.5.1. ERROR REPORT SELECTION MENU

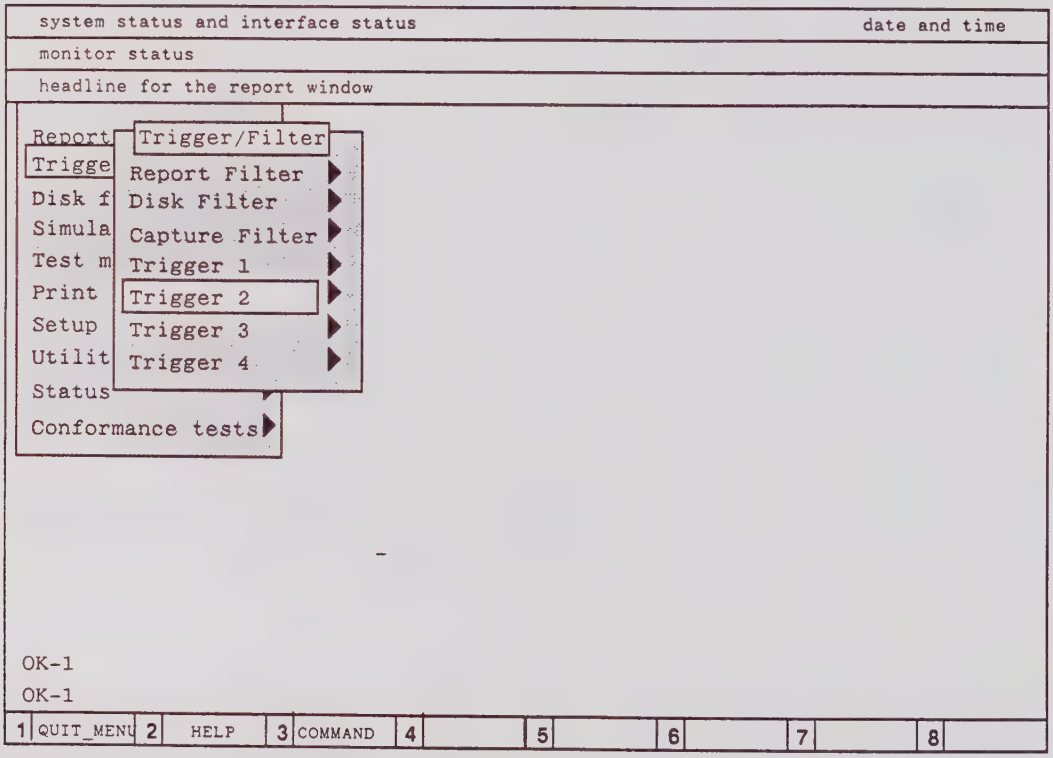
The error report selection menu is only available for primary rate interface. Particular errors on layer 1 are selectable for report. For each error type, an interval of multiframes is to specify in which the error report should be done. The number of detected errors for each selected error type is then displayed on the screen in these intervals



MENU SELECTION:

Bipolar violation	: Select the error report of bipolar violation
CRC-4 error	: Select the error report of CRC-4 errors
Frame error	: Select the error report of frame errors
Multiframe error	: Select the error report of multiframe errors
CEPT Special error	: Select the error report of CEPT special errors

4.6. TRIGGER / FILTER SELECTION MENU

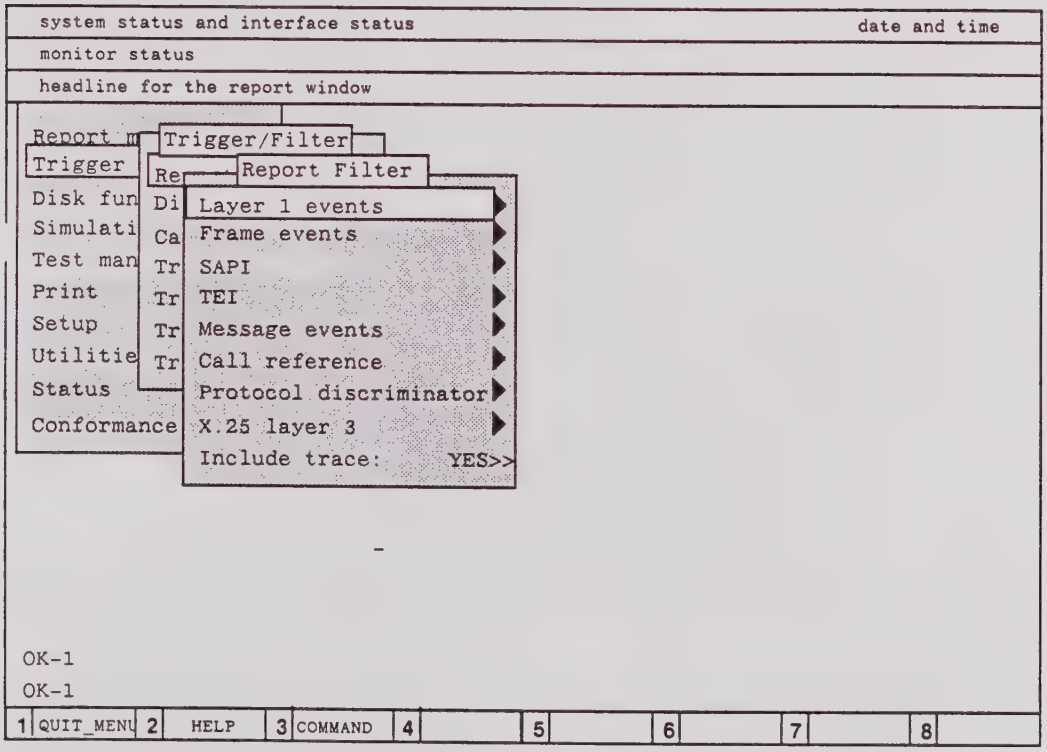


MENU SELECTION:

Report Filter	: Enter report filter selection menu
Disk Filter	: Enter disk filter selection menu
Capture Filter	: Enter capture filter selection menu
Trigger 1	: Enter trigger machine 1 selection menu
Trigger 2	: Enter trigger machine 2 selection menu
Trigger 3	: Enter trigger machine 3 selection menu
Trigger 4	: Enter trigger machine 4 selection menu

4.6.1. FILTER SELECTION MENU

There are three independent filters on the data being processed: Report Display filters, Capture RAM filters and Disk Recording filters. These filters can be set to select the physical layer events, frame or message types. Filters may be set on frames with particular SAPI, TEI, call reference number or protocol discriminator. The filters allow display, capture and storage of only those events which are selected.

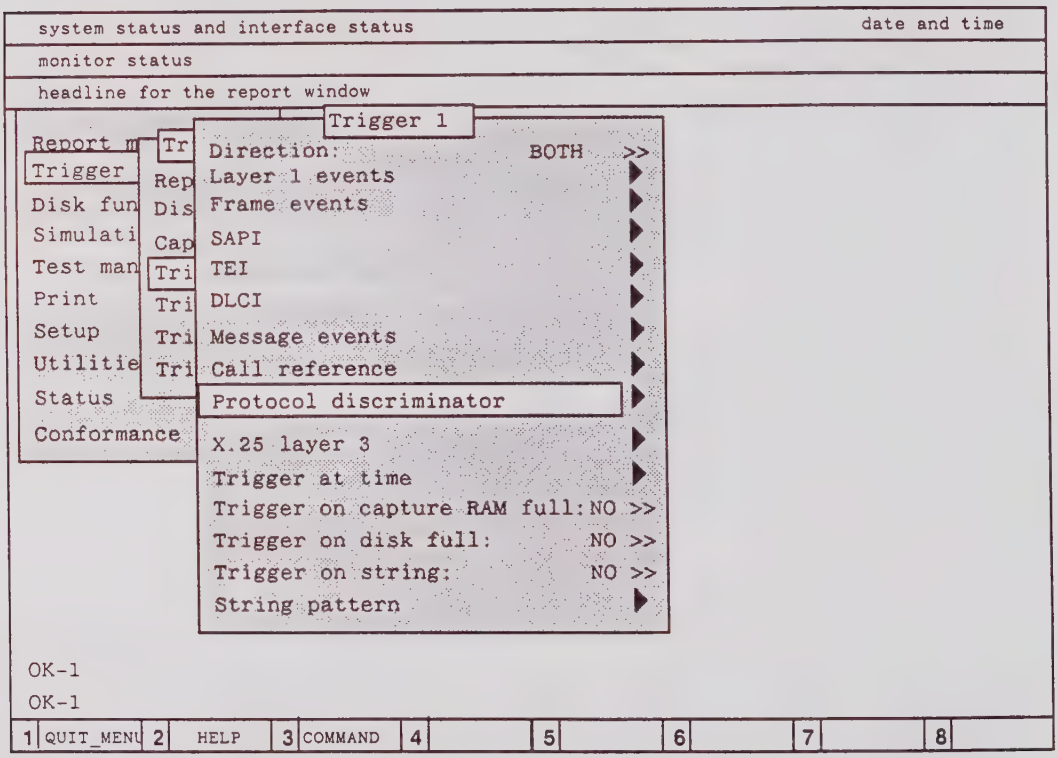


MENU SELECTION:	
Layer 1 events	: Enter menu to select filter on layer 1 events
Frame events	: Enter menu to select filter on layer 2 events
SAPI	: Enter menu to select filter for particular SAPI
TEI	: Enter menu to select filter for particular TEI
Message events	: Enter menu to select filter on layer 3 events
Call reference	: Enter menu to select filter for particular call reference number
Protocol discriminator	: Enter menu to select filter for particular protocol discriminator
X.25 layer 3	: Enter menu to select filter for X.25 data in layer 3
Include trace	: Select filter for trace messages

See chapter 8 for detailed information about filters
and X.25 LAYER 3 D-CHANNEL MANUAL about X.25 data in layer 3.

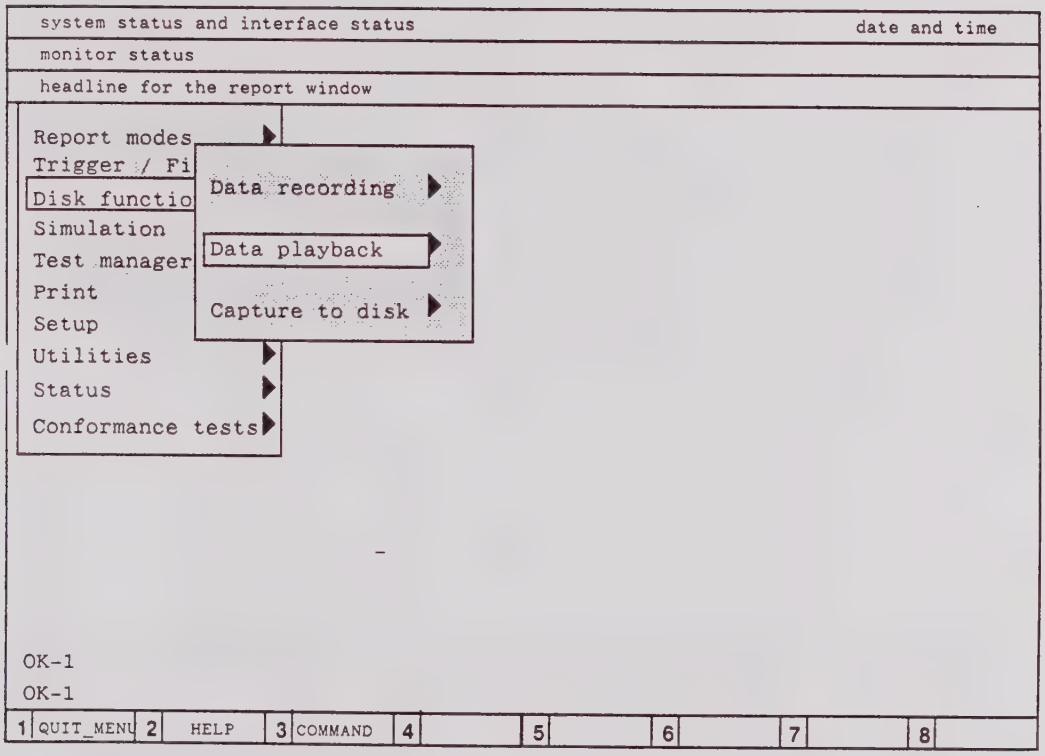
4.6.2. TRIGGER SELECTION MENU

There are four independend trigger machines to allow the user to set trigger and define actions to be performed if that trigger occurs. Triggers can be set on physical layer events, frame or message types as well as on frames with particular SAPI, TEI, call reference number or protocol discriminator. Trigger events can be made origin selective.
(See chapter 9 and X.25 LAYER 3 D-CHANNEL MANUAL for detailed information about triggers .)



MENU SELECTION:	
Direction	: Select the trigger direction
Layer 1 events	: Enter menu to select trigger on layer 1 events
Frame events	: Enter menu to select trigger on layer 2 events
SAPI	: Enter menu to select trigger for particular SAPI
TEI	: Enter menu to select trigger for particular TEI
DLCI	: Enter menu to select trigger for particular SAPI + TEI
Message events	: Enter menu to select trigger on layer 3 events
Call reference	: Enter menu to select trigger for particular call reference number
Protocol discriminator	: Enter menu to select trigger for particular protocol discriminator
X.25 layer 3	: Enter menu to select trigger for X.25 data in layer 3
Trigger at time	: Enter menu to select time trigger
Trigger on capture RAM full	: Select trigger on capture RAM full event
Trigger on disk full	: Select trigger on disk full event
Trigger on string	: Select string trigger
String pattern	: Enter string to be triggered on

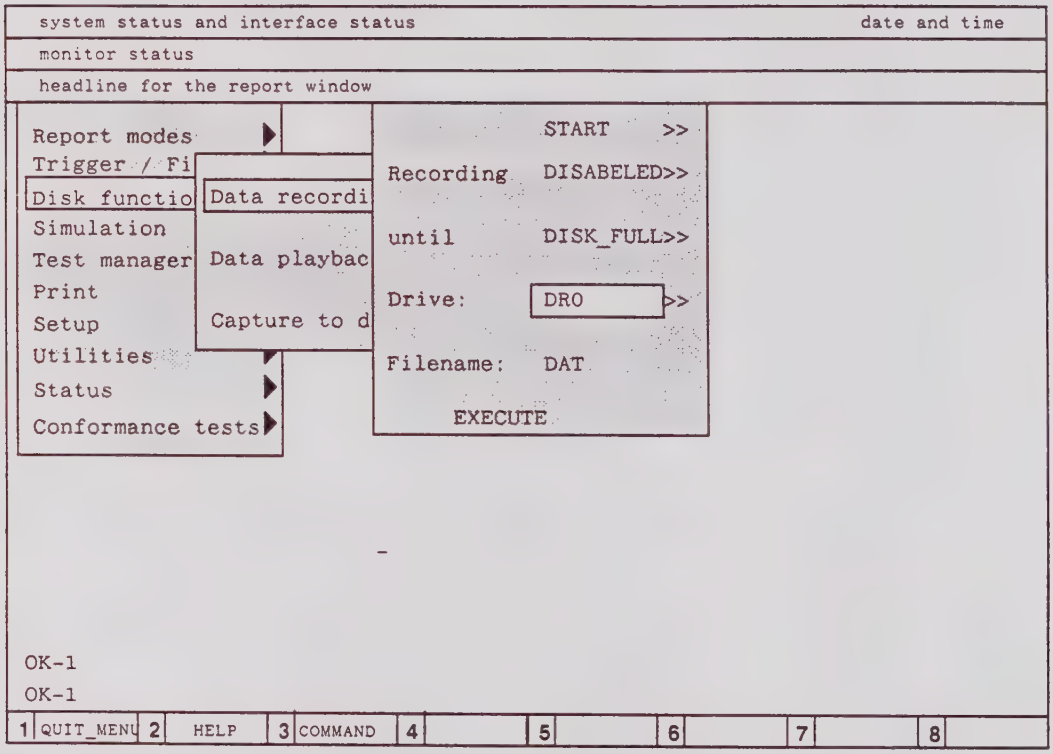
4.7. DISK FUNCTION MENU



MENU SELECTION:

Data recording	: Enter data recording menu
Data playback	: Enter data playback menu
Capture to disk	: Enter menu to copy capture contents to disk

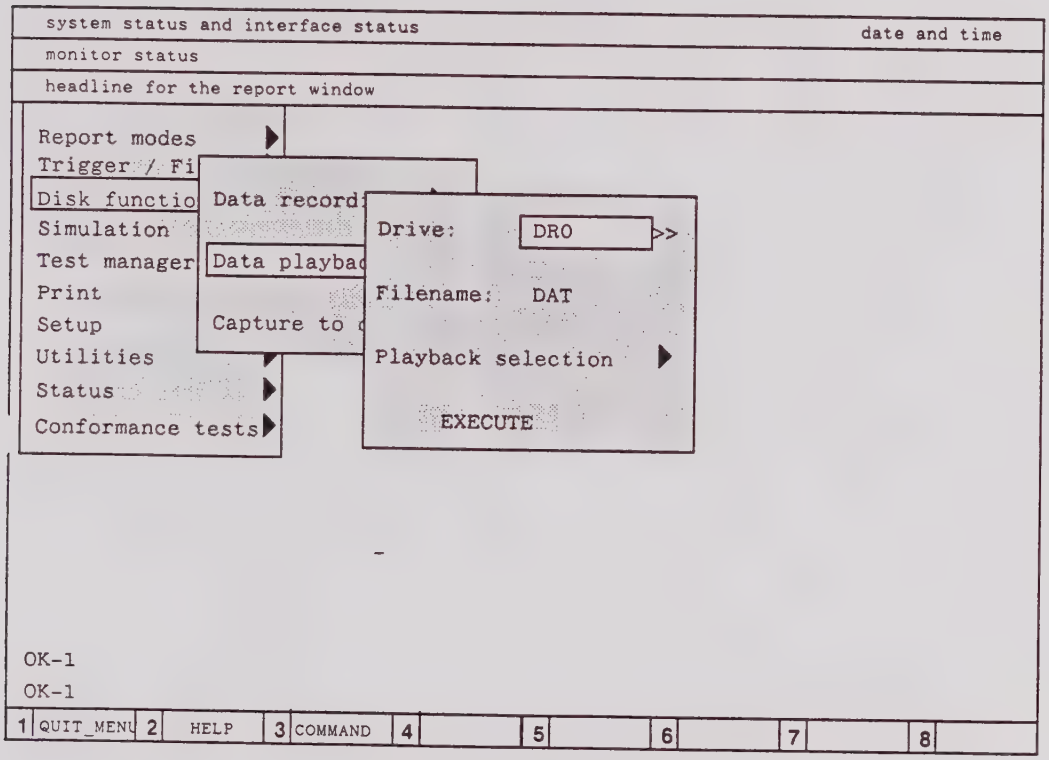
4.7.1. DATA RECORDING MENU



MENU SELECTION:

Recording	Enable or disable data recording
until	: disk is full or disk overwrite
Drive	: Select drive for data recording
Filename	: Name of data recording file
EXECUTE	: Execute start or stop of data recording

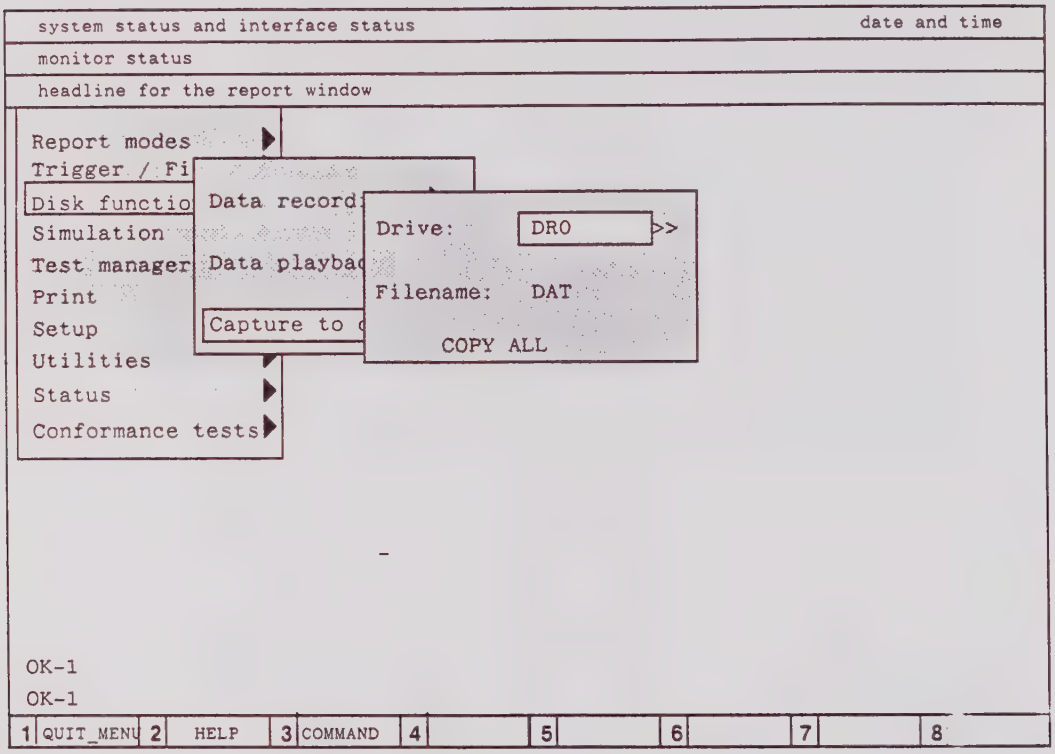
4.7.2. DATA PLAYBACK MENU



MENU SELECTION:

Drive	: Select drive for data playback
Filename	: Name of data playback file
Playback selection	: Enter menu for playback selection
EXECUTE	: Execute data playback

4.7.3. CAPTURE TO DISK MENU



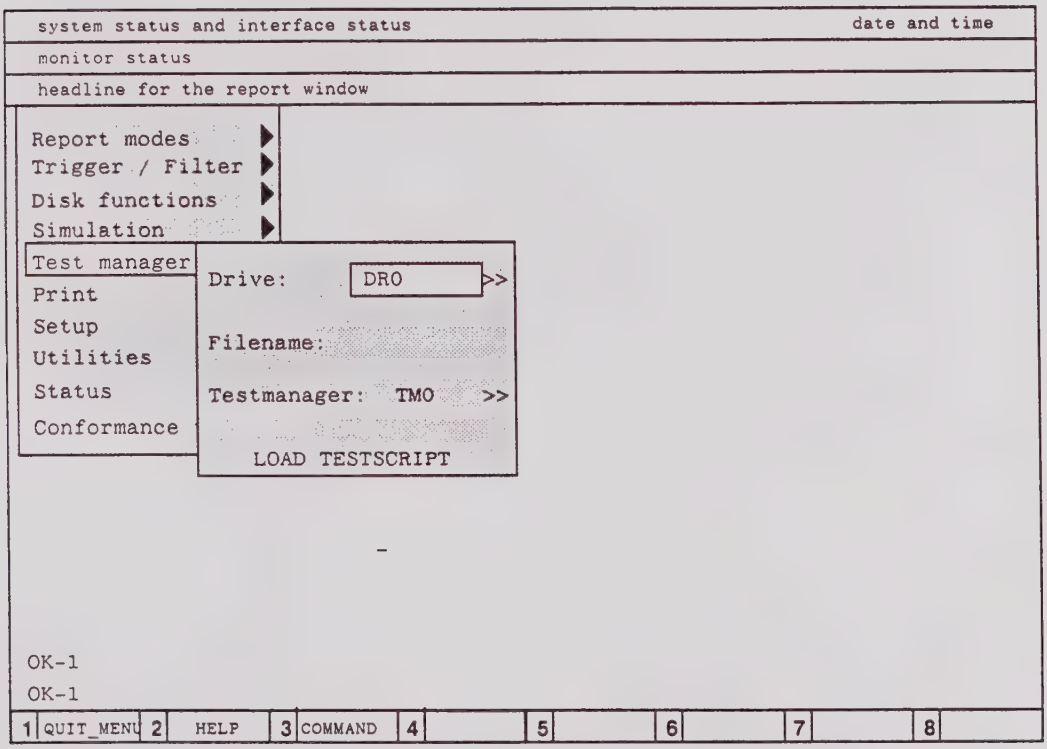
MENU SELECTION:

Drive	: Select drive for data file
Filename	: Name of data file
COPY ALL	: Copy contents of capture RAM to data file

4.8. SIMULATION MENU

See ISDN EMULATION/SIMULATION USER MANUAL.

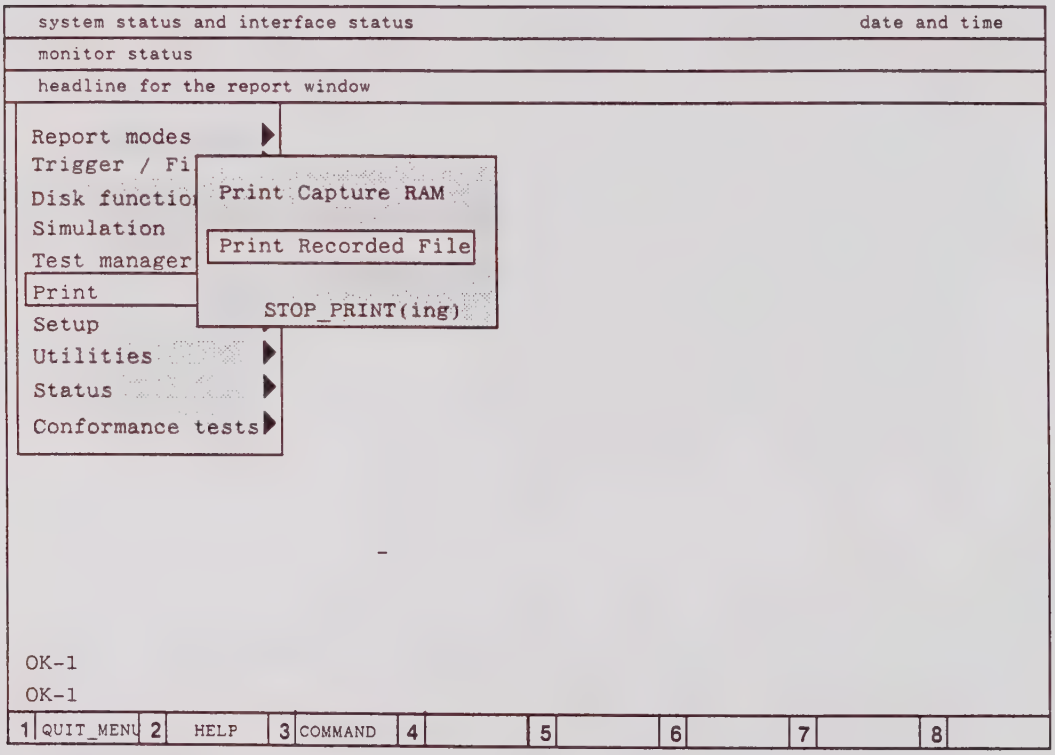
4.9. TEST MANGER SELECTION MENU



MENU SELECTION:

Drive	: Select drive of testmanager file
Filename	: Name of testmanager file
Testmanager	: Select current testmanager
LOAD TESTSCRIPT	: Start loading of testmanager program

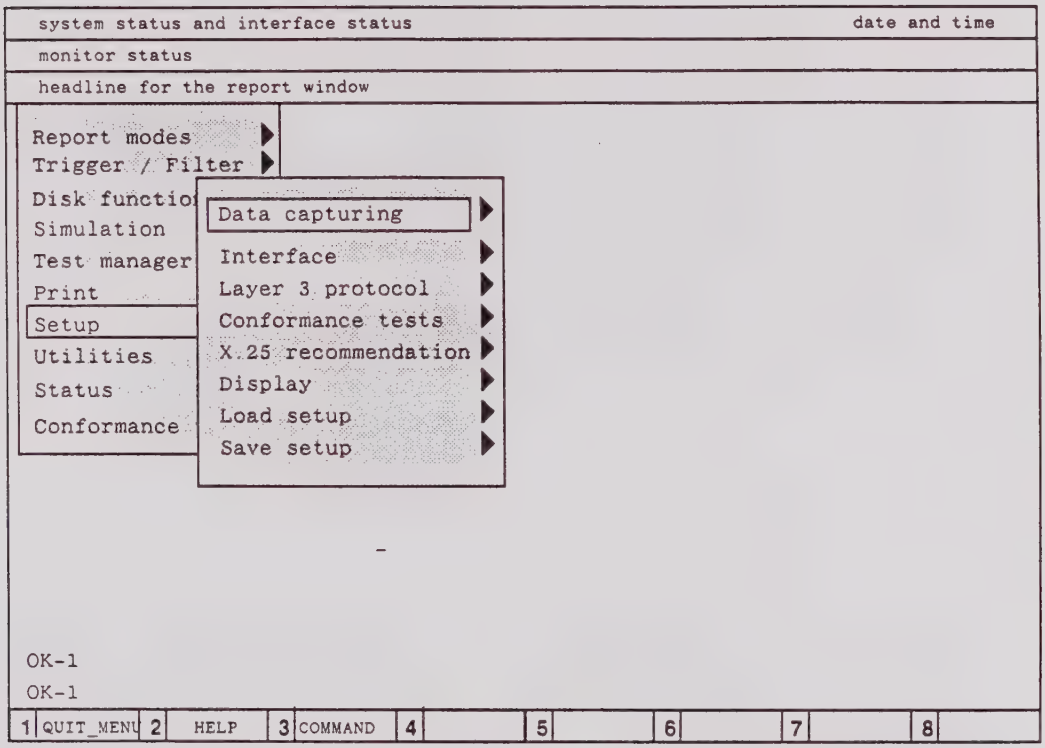
4.10. PRINT MENU



MENU SELECTION:

Print Capture RAM	: Send contents of capture RAM to printer
Print Recorded File	: Send contents of last data recording file to printer
STOP_PRINT(ing)	: Stop printout

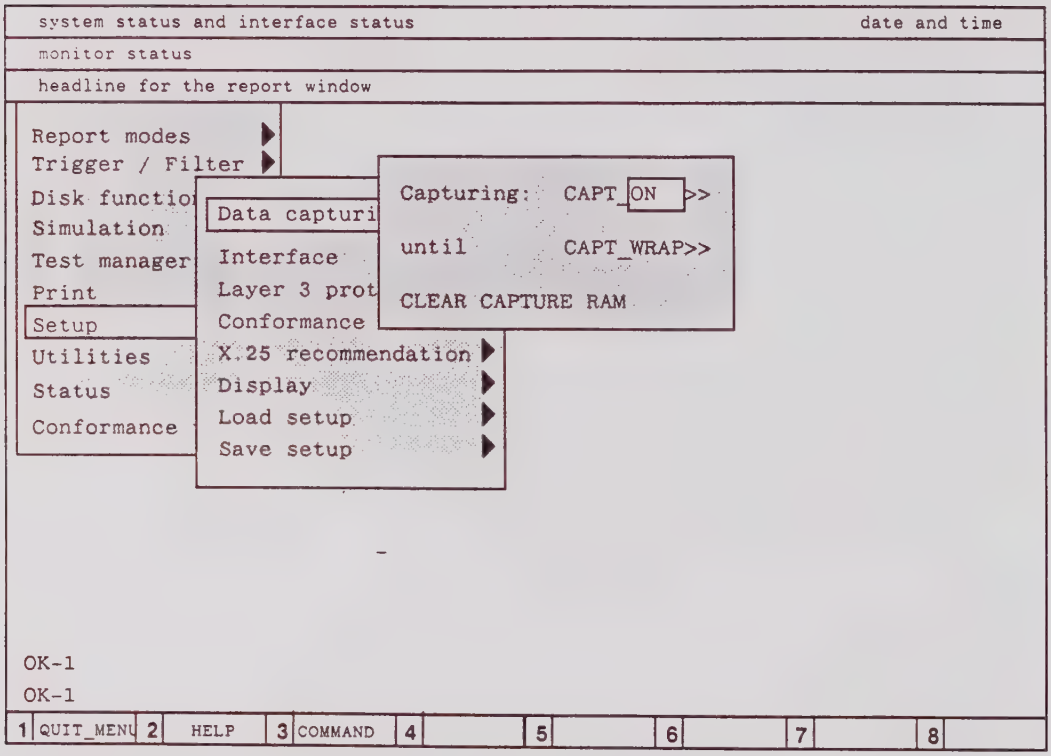
4.11. SETUP MENU



MENU SELECTION:

Data capturing	: Enter menu for capture RAM setup
Interface	: Enter menu for interface setup
Layer 3 protocol	: Enter menu for setup of layer 3 variant
Conformance tests	: Enter menu to load conformance tests (only available in simulator software)
X.25 recommendation	: Enter menu for setup of recommendation of X.25 data in layer 3
Display	: Enter menu for display setup
Load setup	: Enter menu to load a setup file
Save setup	: Enter menu to save a setup file

4.11.1. CAPTURE RAM SETUP MENU

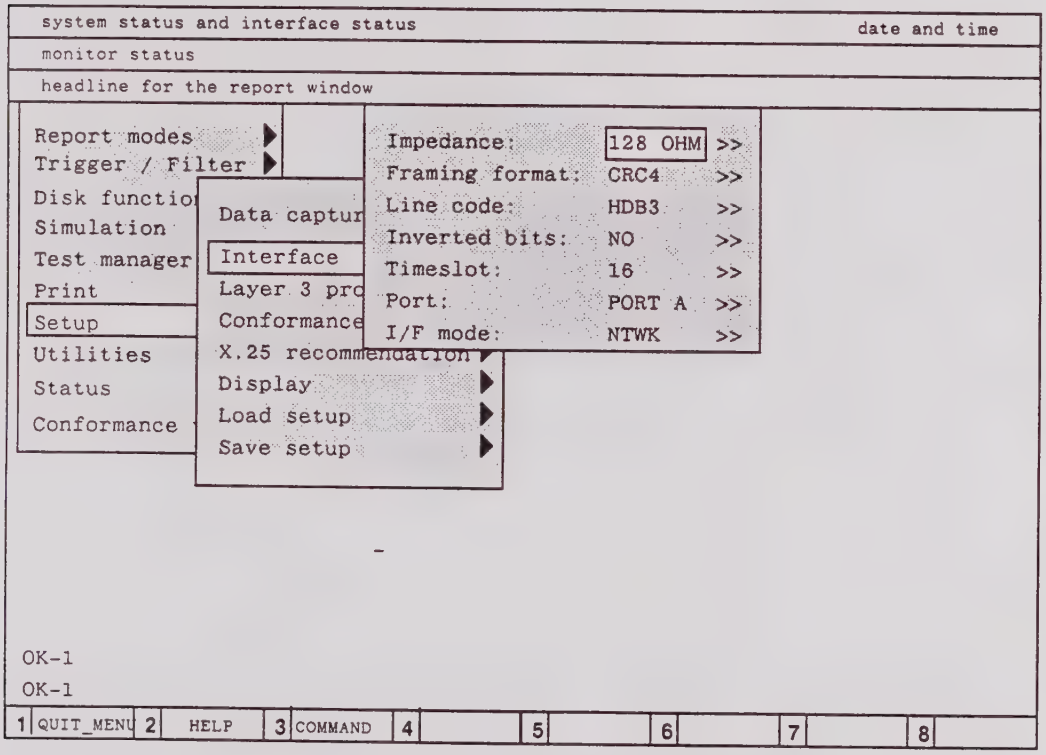


MENU SELECTION:

Capturing	: Select data capturing
until	: Capture RAM is full or capture RAM overwrite
CLEAR CAPTURE RAM	: Delete contents of capture RAM

4.11.2. INTERFACE SETUP MENU (primary rate interface)

The following menu is available for primary rate interface.

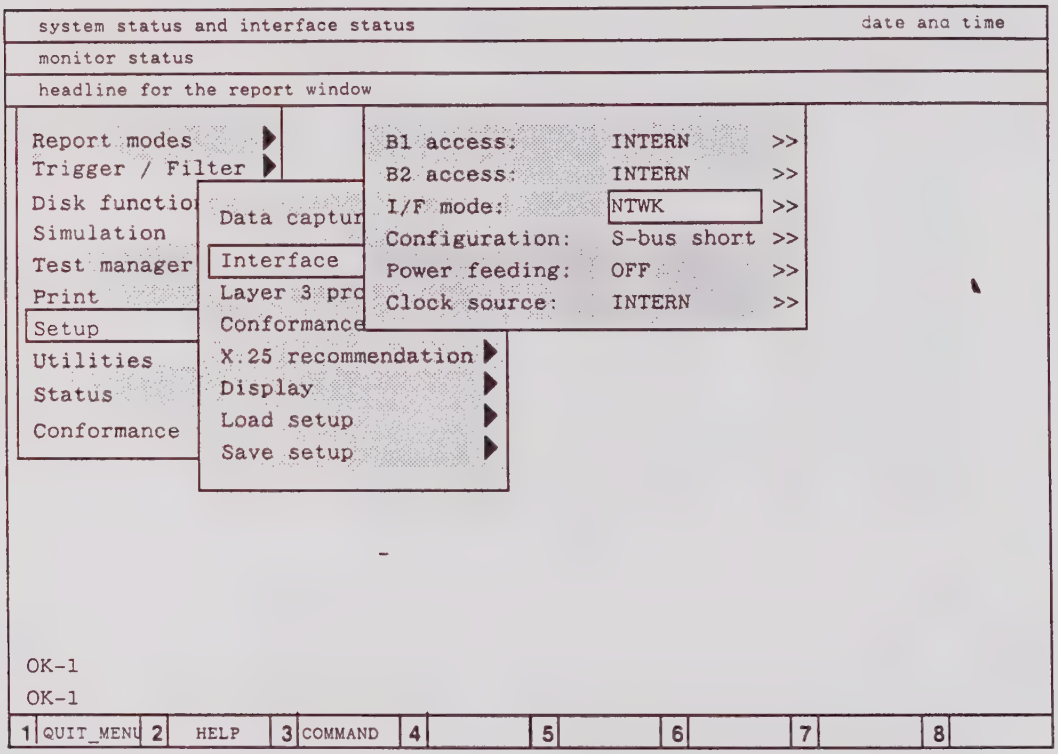


MENU SELECTION:

Impedance	: Select source and sink impedance
Framing format	: Select framing format
Line code	: Select line encoding
Inverted bits	: Select inverted HDLC
Timeslot	: Select timeslot
Port	: Select port for simulation (not available for monitor software)
I/F mode	: Select interface mode for simulator software

4.11.2.1. INTERFACE SETUP MENU (basic access interface)

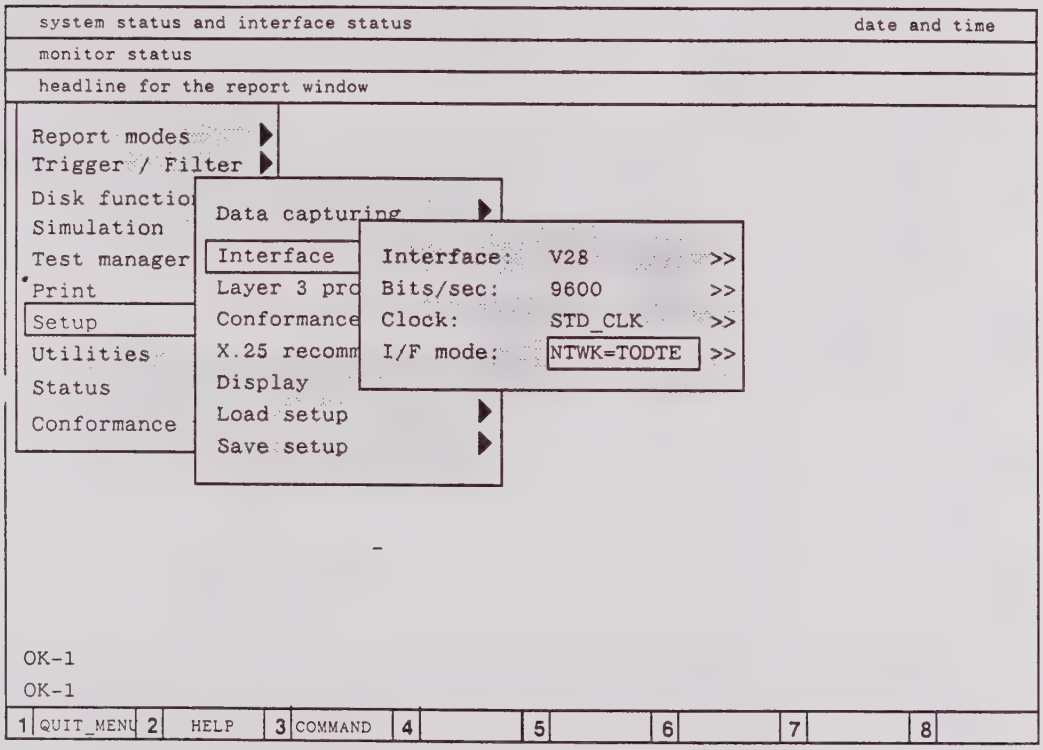
The following menu is available for basic access interface but only for simulator software.



MENU SELECTION:	
B1 access	: Select B1 channel for external access
B2 access	: Select B2 channel for external access
I/F mode	: Select interface mode for emulation/simulation
Configuration	: Select configuration of S-bus for emulation/simulation
Power feeding	: Select power feeding for NT simulation
Clock source	: Select clock source for NT simulation
Impedance	: Select impedance of termination resistors for TE simulation

4.11.2.2. INTERFACE SETUP MENU (V/X interface)

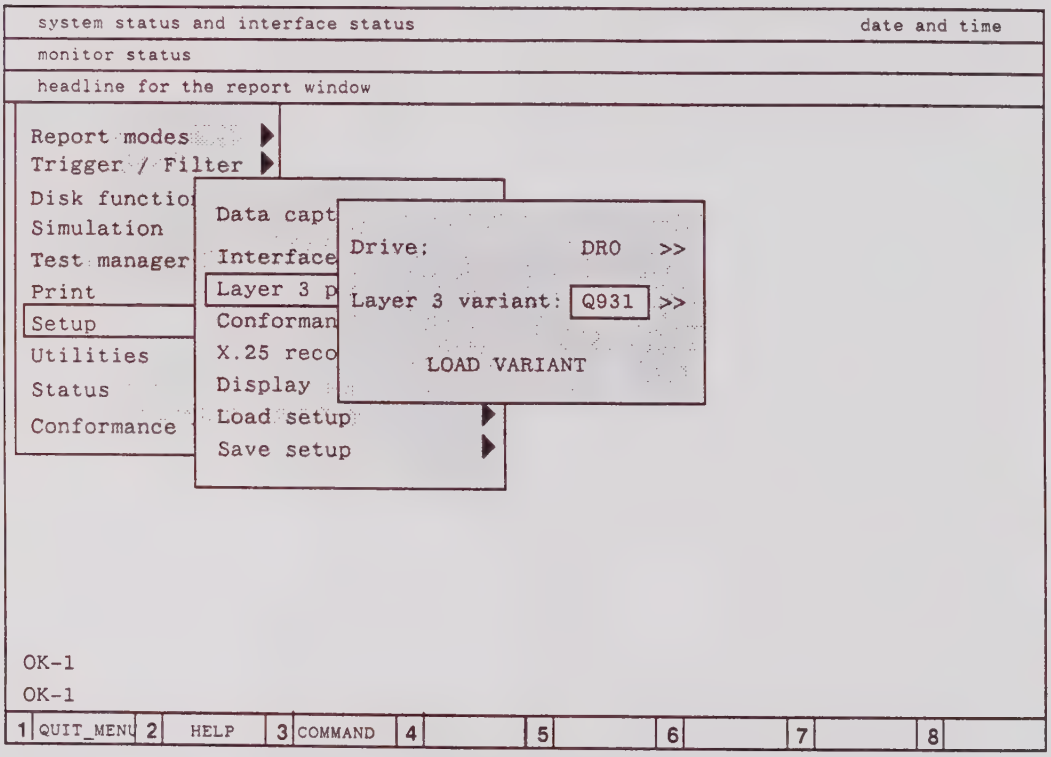
The following menu is available for V/X interface.



MENU SELECTION:

Interface	: Select physical interface
Bits/sec	: Select interface speed for NT simulation
Clock	: Select clocking for NT simulation
I/F mode	: Select interface mode for simulaton

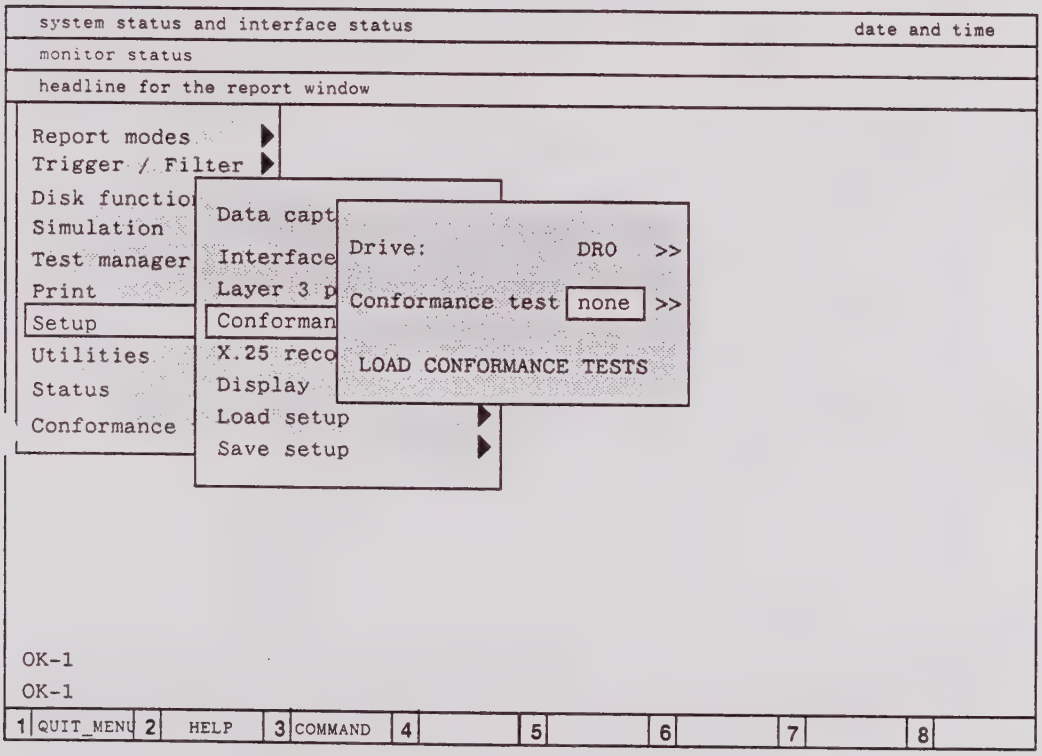
4.11.3. LAYER 3 PROTOCOL SETUP MENU



MENU SELECTION:

Drive	: Select drive of layer 3 variant binary file
Layer 3 variant	: Select variant of layer 3 decoding
LOAD VARIANT	: Execute loading of layer 3 variant

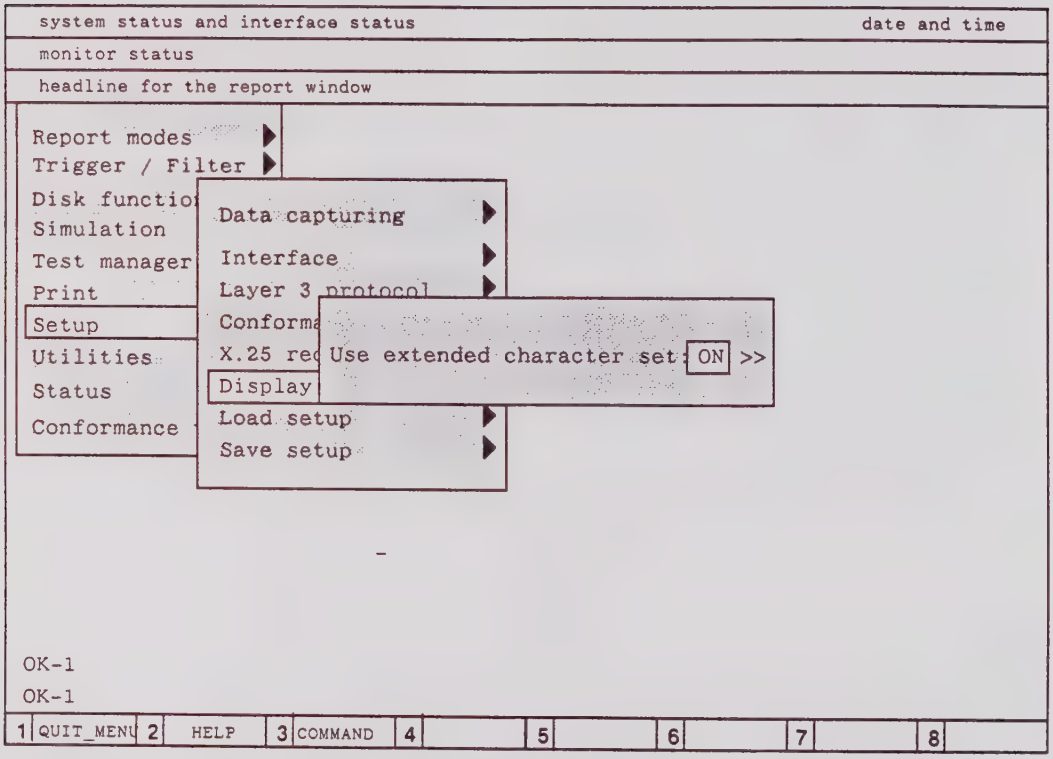
4.11.4. CONFORMANCE TESTS SETUP MENU



MENU SELECTION:

Drive	: Select drive of conformance tests binary file
Conformance test	: Select conformance test packet
LOAD VARIANT	: Execute loading of conformance tests

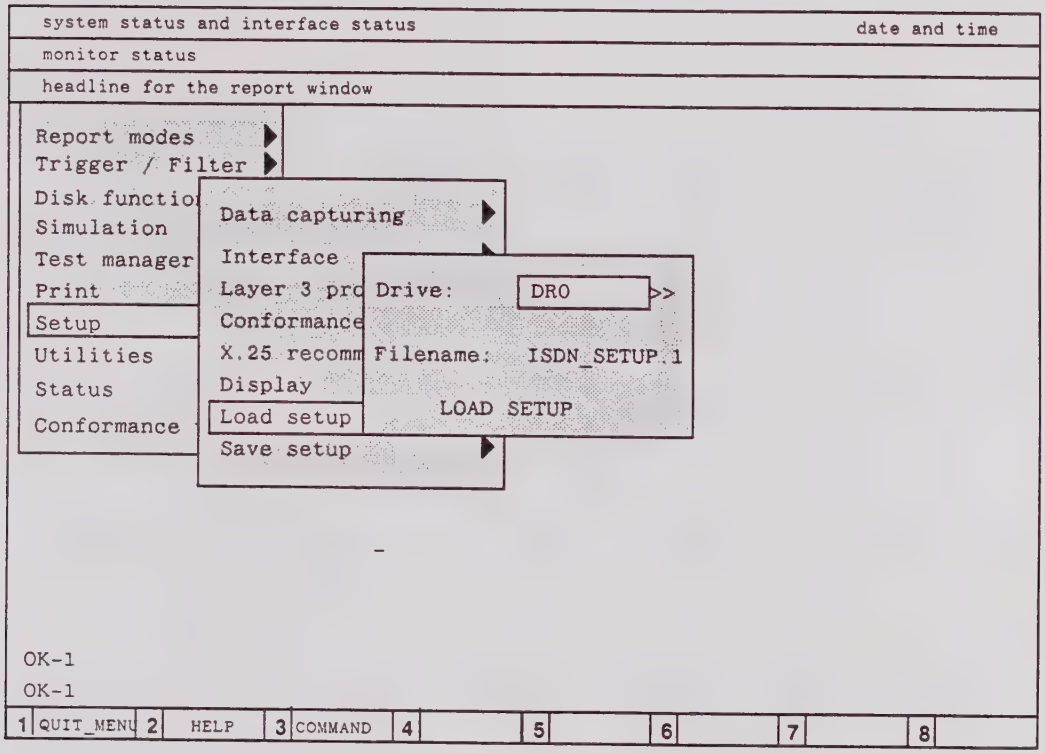
4.11.5. DISPLAY SETUP MENU



MENU SELECTION:

Extended character : Select character set for menu display

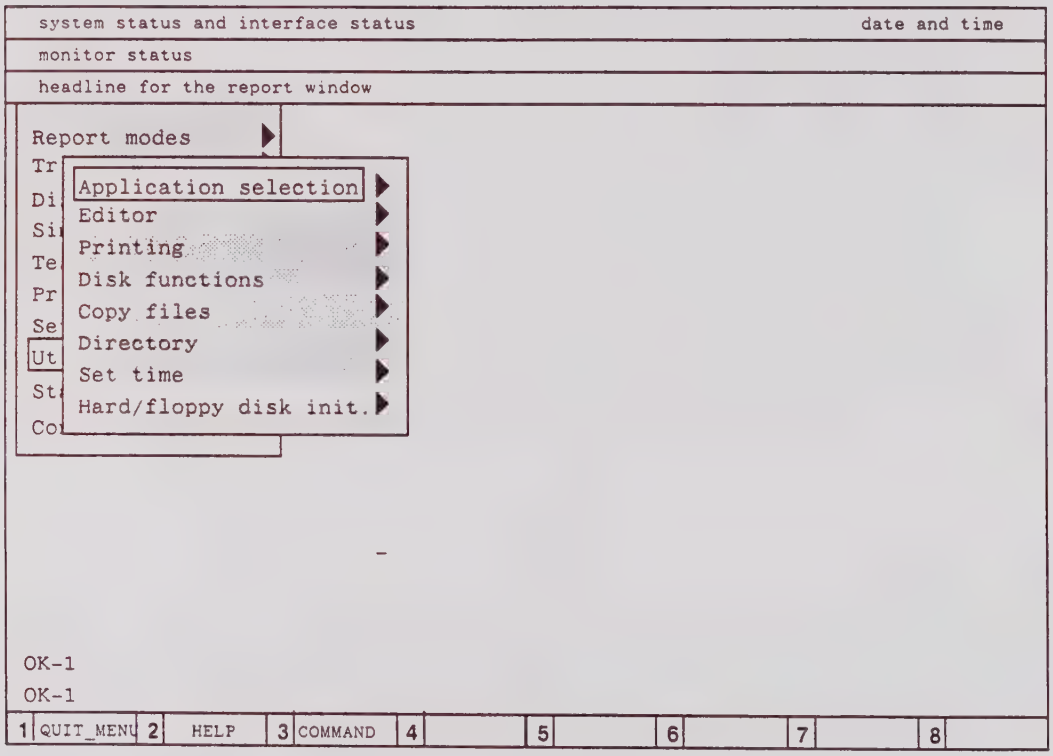
4.11.6. LOAD / SAVE SETUP MENU



MENU SELECTION:

Drive	: Select drive for setup file
Filename	: Name of setup file
LOAD SETUP	: Execute loading of setup file
SAVE SETUP	: Save current configuration in a setup file

4.12. UTILITIES MENU



MENU SELECTION:
(see USER MANUAL)

4.13. STATUS MENU

See ISDN EMULATION/SIMULATION USER MANUAL.

4.14. CONFORMANCE TESTS MENU

See corresponding CONFORMANCE TEST USER MANUAL.

5.4. MESSAGE LEVEL REPORT FORMATS

5.4.1. SHORT REPORT - MSG_SHRT

The short report will display the Layer 3 information up to the message type identifier. The short report is useful for higher speed monitoring because more frames per screen may be displayed and processing time is less than with MSG_MED1/2 or MSG_COMP.

5.4.2. MEDIUM1 REPORT - MSG_MED1

The medium1 report will display the message up to the message type identifier, additional the IE-identifiers will be displayed.

5.4.3. MEDIUM2 REPORT - MSG_MED2

The medium2 report will display the message type identifier, IE identifiers and IE contents. In data display mode MSG_HEX the complete message will be displayed in hex.

5.4.4. COMPLETE REPORT - MSG_COMP

The complete report is the most comprehensive of the message report formats. It will display the complete message in a detailed decode format. The message name and IE's as well as the IE contents are displayed in mnemonic mode. Additionally the message contents is decoded and displayed up to the bit level. Note that Data display commands in MSG_COMP mode are affecting the display of non mnemonic IE contents only.

5.5. MESSAGE DATA DISPLAY MODES

5.5.1. MNEMONIC DISPLAY - MSG_MNEM

The selected message parts are displayed in mnemonic mode.

5.5.2. HEX DISPLAY - MSG_HEX

The selected message parts are displayed in hex mode.

5.5.3. CHAR DISPLAY - MSG_CHAR

The selected message parts are displayed in character mode.

5.5.4. NO DISPLAY - MSG_OFF

Messages are not reported. In MSG_OFF mode the message level report formats are ignored.

5.6. REPORT SCREEN EXAMPLES

The following subsections show examples of display screens for some of the display representations of the ISDN Monitor. The full screens are not shown but a sufficient portion is shown to indicate the type of display to be expected.

5.6.1. Layer 1 Event Display Screens (Examples for Basic access)

L1_OFF

BLOCK_NUMBER	ORIG
0	Usr
1	Net
2	Usr
3	Usr
4	Net
5	Net

C/I_ON and L1_SHRT

BLOCK_NUMBER	ORIG	
0	Usr	Layer 1 Error - UNDEFINED
1	Net	Layer 1 Code = 08 - INFO 2
2	Usr	Layer 1 Code = 08 - INFO 1
3	Usr	Layer 1 Code = 0C - INFO 3
4	Net	Layer 1 Code = 0C - INFO 4
5	Net	Layer 1 is activated

C/I_ON and L1_COMP

BLOCK_NUMBER	ORIG		
0	Usr	Layer 1 Error - not specified by CCITT	
1	Net	Layer 1 - Activation request	INFO 2
2	Usr	Layer 1 - Activation request P8	INFO 1
3	Usr	Layer 1 - Activation indication	INFO 3
4	Net	Layer 1 - Activation indication	INFO 4
5	Net	Layer 1 is activated	

5.6.2. Frame Event Display Screens

FRM_SHRT

BLOCK_NUMBER	ORIG	SAPI	TEI	C/R	FRAME
1	Net	63	127	1	UI
0	Usr				
3	Usr	0	64	0	SABME

FRM_HEX

BLOCK_NUMBER	ORIG	FRM
1	Net	FE FF 03
0	Usr	
3	Usr	00 81 7F

FRM_CHAR

BLOCK_NUMBER	ORIG	FRM
1	Net	~ ± E_X
0	Usr	
3	Usr	N_U S_H ±

FRM_COMP

BLOCK_NUMBER	ORIG	SAPI	TEI	C/R	FRAME	P/F	N(R)	N(S)	
1	Net	63	127	1	UI	P=0			
		Message name							: Identity assigned
		Management entity identifier							: 15
		Reference number (RI)							: 7752
		Message type							: 2
		Action identifier (AI)							: 64
0	Usr								
3	Usr	0	64	0	SABME	P=1			

SPLIT_ON

SAPI	TEI	C/R	FRAME	ORIG	PD	C_REF	MSG_TY	SAPI	TEI	C/R	FRAME	ORIG	PD	C_REF	MSG_TY
								0	64	0	UA				
0	64	0	INFO	Orig	8	4660	SETUP								
								0	64	0	RR				
								0	64	1	INFO	Dest	8	4660	CALL_PR
0	64	1	RR												
								0	64	1	INFO	Dest	8	4660	ALERT
0	64	1	RR												
								0	64	1	INFO	Dest	8	4660	CONN
0	64	1	RR												
0	64	0	INFO	Orig	8	4660	DISC								
								0	64	0	RR				
								0	64	1	INFO	Dest	8	4660	REL

5.6.3. Message Event Display Screens

MSG_SHRT and MSG_MNEM

BLOCK_NUMBER	ORIG	ORIG	PD	C_REF	MSG_TYPE
2	Usr	Orig	Q.931	4660	SETUP
2	Net				
3	Net	Dest	Q.931	4660	CALL_PROC

MSG_MED1 and MSG_MNEM

BLOCK_NUMBER	ORIG	ORIG	PD	C_REF	MSG_TYPE		
2	Usr	Orig	Q.931	4660	SETUP		
BEARER_CAP	CHANNEL_ID	CALLING_NUM	CALLING_SAD	CALLED_NUM			
CALLED_SAD	LOW_LAY_COMP	HI_LAY_COMP	SHIFT	Set(05)/Code(05)			
2	Net						
3	Net	Dest	Q.931	4660	CALL_PROC		
CHANNEL_ID	DISPLAY						

MSG_MED2 and MSG_MNEM

```
BLOCK_NUMBER  ORIG  ORIG  PD     C_REF  MSG_TYPE

 2            Usr   Orig  Q.931  4660   SETUP
                          Bearer capability           : 80 90 A3
                          Channel identification      : A1 A3 8C
                          Calling party number        : 21 80 30 33 35 34 31 36 32 39 32
                          Calling party subaddress    : 80 04
                          Called party number         : A1 30 33 36 30 36 38 33 31 36
                          Called party subaddress     : 80 02
                          Low layer compatibility     : 80 90 A3
                          High layer compatibility    : 91 81
                          Shift                       : 05
                          Set(05)/Code(05)            : D1 81
 2            Net
```

MSG_COMP and MSG_MNEM

```
BLOCK_NUMBER  ORIG

2             Usr
Protocol Discriminator = Q.931 Setup                          Call reference = 4660 Orig
        1   00000100  INFORMATION ELEMENT     : Bearer capability
        2   00000011  IE Length               : 3 Octets
        3   1-------  Extension Bit           : Not Continued
            -00-----  Coding Standard         : CCITT
            ---00000  Info Trans Cap          : Speech
        4   1-------  Extension Bit           : Not Continued
            -00-----  Transfer Mode           : Circuit Mode
            ---10000  Transfer Rate           : 64 kbits/s
        5   1-------  Extension Bit           : Not Continued
            -01-----  Layer Identification    : User Info Layer 1 Protocol
```

Note: The MSG_COMP, MSG_MNEM display format should not be used in the Monitor state for performance reasons.

MSG_SHRT and MSG_HEX

```
BLOCK_NUMBER  ORIG  DATA

 2            Usr   08  02  12  34  05
 2            Net
 3            Net   08  02  92  34  02
```

MSG_MED1 and MSG_HEX

```
BLOCK_NUMBER  ORIG  DATA

  2           Usr   08  02  12  34  05
  04  18  6C  6D  70  71  7C  7D  90  05
  2           Net
  3           Net   08  02  92  34  02
  18  28
  3           Usr
```

MSG_MED2 and MSG_HEX

```
BLOCK_NUMBER  ORIG  DATA

  2           Usr   08  02  12  34  05  04  18  6C  6D  70  71  7C  7D  90  05  A1  02  8E
                    73  91  A6  E9  FF  2E  09  98  33  76  88  26  35  52  32  22  DA  FE
                    56  33  21  DE  FE  A5  6E  87  98  99  34
  2           Net
  3           Net   08  02  92  34  02  18  28  67  88  A3  E4  11  08  E5  76  01  90  12
                    4F  56  10  00  06  56
  3           Usr
  4           Net   08  02  92  34  07
  4           Usr
  5           Net   08  02  92  34  65
  5           Usr
  6           Net   08  02  97  8E  0D  2A  18  09
```

Note: The use of the MSG_CHAR function will produce similar displays to the MSG_HEX displays but in a character set controlled by corresponding commands.

SPLIT_ON

```
SAPI  TEI  C/R  FRAME  ORIG  PD  C_REF  MSG_TY  :  SAPI  TEI  C/R  FRAME  ORIG  PD  C_REF  MSG_TY
                                                :    0    64   0    UA
 0    64   0    INFO   Orig  8   4660   SETUP   :
                                                :    0    64   0    RR
                                                :    0    64   1    INFO   Dest  8   4660   CALL_PR
 0    64   1    RR                              :
                                                :    0    64   1    INFO   Dest  8   4660   ALERT
 0    64   1    RR                              :
                                                :    0    64   1    INFO   Dest  8   4660   CONN
 0    64   1    RR                              :
 0    64   0    INFO   Orig  8   4660   DISC    :
                                                :    0    64   0    RR
                                                :    0    64   1    INFO   Dest  8   4660   REL
```

5.7. REPORT COMMANDS

REP_ON	(--)	The report output to the display window is on.
REP_OFF	(--)	The report output to the display window is off.

NOTE: The following two commands are only available for Basic access interface.

C/I_ON	(--)	The display of commands and indications on the physical layer is on.
C/I_OFF	(--)	The display of commands and indications on the physical layer is off.
L1_SHRT	(--)	Physical layer events reported in short format.
L1_COMP	(--)	Physical layer events reported in complete format.
L1_OFF	(--)	Physical layer events not reported.
FRM_SHRT	(--)	Layer 2 events (Frames) reported in short format.
FRM_COMP	(--)	Layer 2 events reported in complete format.
FRM_OFF	(--)	Layer 2 events not reported.
FRM_HEX	(--)	Layer 2 events displayed in hex format.
FRM_CHAR	(--)	Layer 2 events displayed in character format.
MSG_SHRT	(--)	Layer 3 events (Messages) reported in short format. (i.e. Message name only).
MSG_MED1	(--)	Layer 3 events reported in Medium 1 format. (i.e. Message name and IE name).
MSG_MED2	(--)	Layer 3 events reported in Medium 2 format. (i.e. Message and IE names plus IE contents).
MSG_COMP	(--)	Layer 3 events reported in complete format. (i.e. Complete display and decoding of Messages, IEs, fields and field contents).
MSG_MNEM	(--)	Data Packet field reported according to higher layer rules.
MSG_HEX	(--)	Data Packet field reported in hex format.
MSG_CHAR	(--)	Data Packet field reported in character format.
MSG_OFF	(--)	Data Packet field not reported.

SPLIT_ON	(--)	Split mode of display is turned on. User and Network displays are on seperate halfes of the screen. (Note that the command SPLIT_ON automatically sets the MSG_SHRT)
SPLIT_OFF	(--)	Split mode of display is turned off.
CS=ASCII	(--)	Characters decoded using ASCII codeset.
CS=EBCDIC	(--)	Characters decoded using EDCDIC codeset.
CS=HEX	(--)	Characters decoded using HEX codeset. **Note:** the above (starting with CS..) commands (keys) are used in association with the MSG_CHAR of display.
TIME_ON	(--)	The timestamps for events are reported as seconds, milliseconds and microseconds.
TIME_SEC	(--)	Same as TIME_ON.
TIME_HOUR	(--)	The timestamps for events are reported as hours, minutes, seconds and hundreds of milliseconds.
TIME_YEAR	(--)	The timestamps for events are reported as year, month and day.
BLOCK_ON	(--)	A sequential Block count associated with each event is reported instead of the event timestamps.

5.8. RECEIVER OVERFLOW

The ISDN Monitor will respond with an error message if data is received faster than can be processed. The report display will switch to a faster display mode (REP_SHRT or REP_OFF) when this happens. The shutdown of the display reports can be controlled with the commands **SHUTDOWN_ON** and **SHUTDOWN_OFF** .

SHUTDOWN_ON	(--)	The report format will switch to REP_SHRT and REP_OFF if a receiver overflow occurs.
SHUTDOWN_OFF	(--)	The report format will not change when a receiver overflow occurs.

The following message is displayed, captured and recorded to disk if a receiver overflow occurs:

"RECEIVER OVERFLOW"

6. DISK OPERATION

6.1. OVERVIEW

It is possible to record data onto a file on the floppy disk (or on harddisk if installed). The source of data can be the live data when in either MONITOR or FREEZE state, or from capture ram when in HALT state.

The maximum size of the data file is the size available on the disk. The user can select DISK_FULL, in which case, recording stops when the file is full, or DISK_WRAP, in which case, a continuous recording of the data.

Recording can be done by START recording in the data recording menu (or by the RECORD command) and playback is done by EXECUTE playback in the data playback menu (or by the PLAYBACK command). If no parameter is given, the default filename is used, which may be set by the =TITLE command (See section 6.5 for a description of the commands).

Data is retrieved by EXECUTE PLAYBACK, scrolling control can be done by using the cursor keys (see section 7.3).

The data may be transfered from capture ram to disk when in HALT state.

Note: While data recording is being done on either application, the user must ensure that no other disk operations are attempted on that or any other application. This means that playback cannot be done while the other application is recording data. Also, the disk utilities, editor and directory functions should not be used while any application is recording data.

6.2. MULTIPORT DATA RECORDING

Recording data from more than one application into the same data file can be done. The data filenames and disk drive specified must be the same on all applications where disk recording should be done (ISDN applications and B-channel applications as far as available). The procedure would be to create the data file on one application by START recording, switch to another application and START recording there. The data file will then contain the data from all applications where disk recording is active. The recording must be closed by STOP recording from the data recording menu (or doing the DISK_OFF command) in all applications.

Either application can be used to playback the file by EXECUTE playback in the data playback menu (or using the PLAYBACK command), then using the cursor keys to scroll through the recorded data.

In the playback selection menu (or with the PLAY_1, PLAY_2, ... PLAY_7 and PLAY=ON and PLAY=OFF commands), the viewing of each application's data can be controlled.

Notes: - Data recorded from the own application is displayed in green/white.

- Data recorded from other applications is displayed in yellow/cyan.

- By default, all recorded data will be displayed.

6.3. MULTPORT PLAYBACK SELECTION

To see or suppress display of data recorded from other applications, enter the data playback selection menu and select the data which should be displayed.

Notes: – Display from Capture Ram uses the same mechanisms as Playback, be sure to have set PLAY=ON for your local application to see the contents of the Capture Ram.

– When data recorded from other applications is displayed error messages may be displayed (such as "INV" , "UNKNOWN" etc.), because the data may have been recorded from an application different from the application doing the playback.

6.4. PLAYBACK SELECTION COMMANDS

PLAY_1 (--)
Select D-channel application (PRI interface) or first B-channel application (BAI interface) of interface 1 to switch playback of data from disk files from this application on or off.

PLAY_2 (--)
Select (second) B-channel application of interface 1 to switch playback of data from disk files from this application on or off.

PLAY_3 (--)
Select D-channel application (PRI interface) or first B-channel application (BAI interface) of interface 2 to switch playback of data from disk files from this application on or off.

PLAY_4 (--)
Select (second) B-channel application of interface 2 to switch playback of data from disk files from this application on or off.

PLAY_7 (--)
Select D-channel application (BAI interface) of interface 1 to switch playback of data from disk files from this application on or off.

PLAY_8 (--)
Select D-channel application (BAI interface) of interface 2 to switch playback of data from disk files from this application on or off.

PLAY=ON (--)
Turn on display of data from selected application when doing playback of data files.

PLAY=OFF (--)
Turn off display of data from selected application when doing playback of data files.

6.5. DISK COMMANDS

RECORD (--)
Opens a data recording file on disk. Use the form RECORD "FILENAME" to specify the filename of the record. (Don't specify the filename within "!). Use the commands **DR0..WD7** prior to the RECORD command to specify the disk drive.

PLAYBACK (--)
Opens an existing data file on disk for playback. Use the command PLAYBACK "FILENAME" to specify the playback file. The FORWARD, TOP, BOTTOM, SCRN_FWD and SCRN_BWD commands or the cursor keys can be used to scroll forward or backward through the playback file with the current selections of the report commands determining the display format.

=TITLE (--)
Enters the disc record or playback title in the form "TITLE" =TITLE . The new title is displayed on the status line.

DISK_OFF (--)
The current Disk recording or playback file is closed. This **must** be done **before** opening another file for playback and **after** recording is finished.

ENB_REC (--)
Live data received is saved in a disk recording when in Monitor or Freeze states and a data recording file is open. Data can also be transferred from capture to disk in the Halt state when the CTOD_ON is selected.

DIS_REC (--)
The data is not saved on disk.

FROM_CAPT (--)
Capture ram is selected as source of data when in Halt state.

FROM_DISK (--)
Disk playback file is selected as source of data when in Halt state.

NORM_MODE (--)
Live data received will be displayed, captured and saved in a disk recording file (if selected) in normal mode (on the contrary to **FAST_MODE**).

FAST_MODE (--)
The fast mode is useful for monitoring of high data transfer.
Live data received is saved in a disk recording file when in Monitor or Freeze state and a data recording file is open. The data will not be saved in the capture buffer, even if CAPT_ON is selected. The screen display of the received data is reduced (depending on the speed of incoming data).

7. COMMANDS

7.1. GENERAL COMMANDS

DR0 (--)
Disk drive 0 is selected for recording or playback.

DR1 (--)
Disk drive 1 is selected for recording or playback.

WD0.. (--)
Hard disk partition 0.. is selected for recording or playback.
(for Hard disk option only)

CAPT_ON (--)
The data is saved in capture ram when in Monitor state.

CAPT_OFF (--)
The data is not saved in capture ram.

CAPT_WRAP (--)
Capture buffer initialized with overwrite selected.

CAPT_FULL (--)
Capture buffer initialized with overwrite off. The capture will continue until the buffer is full, then stop.

CLEAR_CRT (--)
The display screen is cleared.

HALT (--)
Enter the HALT mode of the ISDN Monitor.

FREEZE (--)
Enter the FREEZE mode of the ISDN Monitor.

MONITOR (--)
Enter the MONITOR mode of the ISDN Monitor.

7.2. TRANSFER COMMANDS

Transfer commands are used in the transfer of data between the disk and capture RAM.

TRANSFER (--)
Data is transferred from capture RAM or disk as selected by the FROM_CAPT or FROM_DISK commands. This command is provided for capture RAM to disk recording transfer or for capture ram or disk playback when a printout is required.

QUIT_TRA (--)
Quits the transfer prior to finishing.

TRA_ALL (--)
Select everything out of the capture RAM or currently open playback file to be printed or copied to disk (HALT and FREEZE mode only).

TRA_START (--)
Select the startpoint, from where data is to to be printed or copied to disk (HALT and FREEZE mode only).

TRA_END (--)
Select the endpoint, from where data is to to be printed or copied to disk (HALT and FREEZE mode only).

CTOD_ON (--)
Transfer of data from capture to disk is on when in Halt state.

CTOD_OFF (--)
Transfer of data from capture to disk is off.

PRINT_ON (--)
Display reports are printed.

PRINT_OFF (--)
Display reports are not printed.

7.3. SCROLL CONTROL FOR CAPTURE AND DISK PLAYBACK

FORWARD or **F** or	Down arrow	– Scroll forward one line
BACKWARD or **B** or	Up arrow	– Scroll backward one line
SCRN_FWD or **FF** or	CTRL down arrow	– Scroll forward one screen
SCRN_BACK or **BB** or	CTRL up arrow	– Scroll backward one screen
TOP or	SHIFT CTRL up arrow	– Scroll one screen from top
BOTTOM or	SHIFT CTRL down arrow	– Scroll one screen from end

7.4. B-CHANNEL ACCESS

For the external access of a B-channel via the BRI or PRI interface board, the following commands are available:

EXT_BCH_ON **(B-channel number --)**
Configures the connector module that an external access to the specified B-channel is possible.

EXT_BCH_OFF **(--)**
Configures the connector module that an external access to a B-channel is no longer possible.

7.5. CONFIGURATION OF THE PRIMARAY RATE INTERFACE

The following commands are only available for ISDN software on a primary rate interface.

FFMT=CRC4	(--)	Set frame format to CRC4 (for S2M connector)
FFMT=PCM	(--)	Set frame format to PCM (for S2M connector)
FFMT=ESF	(--)	Set frame format to ESF (for T1 connector)
FFMT=D4	(--)	Set frame format to D4 (for T1 connector)
LCOD=AMI	(--)	Set line encoding to AMI
LCOD=HDB3	(--)	Set line encoding to HDB3 (for S2M connector)
LCOD=B8ZS	(--)	Set line encoding to B8ZS (for T1 connector)
IMPED=75	(--)	Set source and sink impedance to 75 ohm (for S2M connector)
IMPED=120	(--)	Set source and sink impedance to 120 ohm (for S2M connector)
IMPED=HI	(--)	Set source and sink impedance to high (for S2M connector)
LENGTH=0	(--)	Set line length to 0-220 ft. (for T1 connector)
LENGTH=2	(--)	Set line length to 440-655 ft. (for T1 connector)
IHDLC_ON	(--)	Switch on reception and transmission of inverted HDLC
IHDLC_OFF	(--)	Switch off reception and transmission of inverted HDLC
=TSLOT	(timeslot --)	Assign the timeslot for monitoring channel

7.6. CONFIGURATION OF THE V/X INTERFACE

The following commands are only available for ISDN software on a V/X interface.

I/F=V.28	(--)	Set interface type to V.28
I/F=V.11	(--)	Set interface type to V.11
I/F=V.35	(--)	Set interface type to V.35
I/F=V.36	(--)	Set interface type to V.36

8. PROGRAMMABLE FILTERS

The ISDN Monitor incorporates programmable filters which allow the user to pass selected events and to filter out others. This means that unnecessary processing is avoided when monitoring at higher data rates, only relevant information is displayed on the screen and unwanted information, such as RR polling, is not stored in the Capture RAM or on the Disk. The ability to filter the data received according to a specific source and destination is also provided.

There are three seperate filters which can act independently of each other. These are called the Report, Capture and Disk filters as they selectively pass information to the Display, to the Capture RAM and to the Disk respectively.

As the filters are independent, it is possible to have different amounts of information going to the Display, the Capture RAM and the Disk at the same time.

For example, the following commands result in the display of only the SABME and UA Frames.

R2=NONE (No frames passed through the report filter)
R2+SABME (SABME frames passed to the Report Generator)
R2+UA (UA frames passed to the Report Generator)

Note: Triggers have precedence over Filters. This means that even though a message has been filtered out and does not appear on the report screen, it can still cause a trigger action to be executed.

8.1. REPORT FILTERS

The Report Filters pass only selected events to the report generator. Thus, the user may select specified frames to be reported while all others are ignored.

The following commands allow selective frame type filtering before the data is passed to the Report Generator.

8.1.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS)

R1=ALL	(--)	All layer 1 events wanted for report
R1=NONE	(--)	No layer 1 events wanted for report
R1+ACTIVATED	(--)	ACTIVATION wanted for report
R1+DEACTIVATED	(--)	DEACTIVATION wanted for report
R1+LSTFRM	(--)	LOST FRAMING wanted for report
R1+RECOVERY	(--)	RECOVERY wanted for report
R1+RI2ERR	(--)	Too many RCV ERROR wanted for report
R1+UNDEFINED	(--)	CCITT UNDEFINED ERROR wanted for report
R1-ACTIVATED	(--)	ACTIVATION not wanted for report
R1-DEACTIVATED	(--)	DEACTIVATION not wanted for report
R1-LSTFRM	(--)	LOST FRAMING not wanted for report
R1-RECOVERY	(--)	RECOVERY not wanted for report
R1-RI2ERR	(--)	Too many RCV ERROR not wanted for report
R1-UNDEFINED	(--)	CCITT UNDEFINED ERROR not wanted for report

8.1.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS)

R1=ALL	(--)	All layer 1 events wanted for report
R1=NONE	(--)	No layer 1 events wanted for report
R1+RED_ALARM	(--)	RED ALARM wanted for report
R1+YEL_ALARM	(--)	YELLOW ALARM wanted for report
R1+LST_SIGL	(--)	LOST OF SIGNAL LEVEL wanted for report
R1+LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK wanted for report
R1+NORMAL	(--)	BACK TO NORMAL wanted for report
R1+BPVL	(--)	BIPOLAR VIOLATION wanted for report
R1+CRCR	(--)	CRC ERROR wanted for report
R1+MERR	(--)	MULTIFRAME ERROR wanted for report
R1+FERR	(--)	FRAME ERROR wanted for report
R1+CEPTSPEC	(--)	CEPT SPECIAL ERROR wanted for report
R1-RED_ALARM	(--)	RED ALARM not wanted for report
R1-YEL_ALARM	(--)	YELLOW ALARM not wanted for report
R1-LST_SIGL	(--)	LOST OF SIGNAL LEVEL not wanted for report
R1-LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK not wanted for report
R1-NORMAL	(--)	BACK TO NORMAL not wanted for report
R1-BPVL	(--)	BIPOLAR VIOLATION not wanted for report
R1-CRCR	(--)	CRC ERROR not wanted for report
R1-MERR	(--)	MULTIFRAME ERROR not wanted for report
R1-FERR	(--)	FRAME ERROR not wanted for report
R1-CEPTSPEC	(--)	CEPT SPECIAL ERROR not wanted for report

8.1.3. FRAME LEVEL FILTER COMMANDS

R2=ALL	(--)	All frame types selected for report
R2=NONE	(--)	No frames selected for report
R2+SABME	(--)	SABME wanted for report
R2+DISC	(--)	DISC wanted for report
R2+UA	(--)	UA wanted for report
R2+DM	(--)	DM wanted for report
R2+FRMR	(--)	FRMR wanted for report
R2+INV	(--)	Invalid frame wanted for report
R2+I	(--)	I frame wanted for report
R2+UI	(--)	UI wanted for report
R2+RR	(--)	RR wanted for report
R2+RNR	(--)	RNR wanted for report
R2+REJ	(--)	REJ wanted for report
R2+XID	(--)	XID wanted for report
R2-SABME	(--)	SABME not wanted for report
R2-DISC	(--)	DISC not wanted for report
R2-UA	(--)	UA not wanted for report
R2-DM	(--)	DM not wanted for report
R2-FRMR	(--)	FRMR not wanted for report
R2-INV	(--)	Invalid frame not wanted for report
R2-I	(--)	I frame not wanted for report
R2-UI	(--)	UI not wanted for report
R2-RR	(--)	RR not wanted for report
R2-RNR	(--)	RNR not wanted for report
R2-REJ	(--)	REJ not wanted for report
R2-XID	(--)	XID not wanted for report
=RF_TEI	(TEI --)	Selects the TEI value which shall be reported.
<>RF_TEI	(TEI --)	Selects the TEI value which shall be ignored.
RF_TEI=ALL	(--)	Selects all TEI values for report.
RF_TEI=127	(--)	Selects additionally TEI 127 for report when another TEI value is already chosen (not 127). Otherwise there is no action.
RF_TEI<>127	(--)	Deselects TEI 127 if it was chosen by **RF_TEI=127**. Otherwise there is no action.

=RF_SAPI (SAPI --) Selects the SAPI value which shall be reported.

<>RF_SAPI (SAPI --) Selects the SAPI value which shall be ignored.

RF_SAPI=ALL (--) Selects all SAPI values for report.

8.1.4. MESSAGE LEVEL FILTER COMMANDS

Messages are added or removed from the Report filter list by initially entering the command M#ABCDEF, where ABCDEF represents the mnemonic string for a given message type. (For example, M#SETUP or M#SETUP_ACK). See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual part for a complete list of Layer 3 message identifiers. Thereafter the nominated message is added (**R+MSG**) or removed (**R-MSG**) from the Report filter list. Note that adding a message that has already been added or removing one that has already been removed has no effect.

Invalid messages and undefined messages are specified using **M#INVALID** and **M#UNDEF** respectively. They are added or removed from the filter list in the same way as other messages.

Protocol discriminator and call reference values can be altered or added to the Report filters with the following FORTH commands:

8.1.4.1. PROTOCOL DISCRIMINATOR

=F_P_DISC **(PD value --)**
Add the Protocol Discriminator event with the given PD value to the active filter list.

-F_P_DISC **(--)**
Remove the Protocol discriminator event from the active filter list.

8.1.4.2. CALL REFERENCE

=F_C_REF **(CR value --)**
Add the Call Reference event with the given CR value to the active filter list.

-F_C_REF **(--)**
Remove the Call Reference event from the active filter list.

For Example,

```
0x0F =F_P_DISC
4660 =F_C_REF
```

Note: The order in which filtering is performed is:
1. Protocol Discriminator, 2. Call Reference, 3. Message

8.2. CAPTURE FILTERS

The Capture Filters will pass only selected frames to the Capture buffer for storage. Thus, the user may select specified frames to be captured and all others ignored.

The following commands allow selective frame type filtering before the data is passed to the Capture RAM.

8.2.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS)

C1=ALL	(--)	All layer 1 events wanted for capture
C1=NONE	(--)	No layer 1 events wanted for capture
C1+ACTIVATED	(--)	ACTIVATION wanted for capture
C1+DEACTIVATED	(--)	DEACTIVATION wanted for capture
C1+LSTFRM	(--)	LOST FRAMING wanted for capture
C1+RECOVERY	(--)	RECOVRY wanted for capture
C1+RI2ERR	(--)	Too many RCV ERROR wanted for capture
C1+UNDEFINED	(--)	CCITT UNDEFINED ERROR wanted for capture
C1-ACTIVATED	(--)	ACTIVATION not wanted for capture
C1-DEACTIVATED	(--)	DEACTIVATION not wanted for capture
C1-LSTFRM	(--)	LOST FRAMING not wanted for capture
C1-RECOVERY	(--)	RECOVERY not wanted for capture
C1-RI2ERR	(--)	Too many RCV ERROR not wanted for capture
C1-UNDEFINED	(--)	CCITT UNDEFINED ERROR not wanted for capture

8.2.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS)

C1=ALL	(--)	All layer 1 events wanted for capture
C1=NONE	(--)	No layer 1 events wanted for capture
C1+RED_ALARM	(--)	RED ALARM wanted for capture
C1+YEL_ALARM	(--)	YELLOW ALARM wanted for capture
C1+LST_SIGL	(--)	LOST OF SIGNAL LEVEL wanted for capture
C1+LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK wanted for capture
C1+NORMAL	(--)	BACK TO NORMAL wanted for capture
C1+BPVL	(--)	BIPOLAR VIOLATION wanted for capture
C1+CRCR	(--)	CRC ERROR wanted for capture
C1+MERR	(--)	MULTIFRAME ERROR wanted for capture
C1+FERR	(--)	FRAME ERROR wanted for capture
C1+CEPTSPEC	(--)	CEPT SPECIAL ERROR wanted for capture
C1-RED_ALARM	(--)	RED ALARM not wanted for capture
C1-YEL_ALARM	(--)	YELLOW ALARM not wanted for capture
C1-LST_SIGL	(--)	LOST OF SIGNAL LEVEL not wanted for capture
C1-LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK not wanted for capture
C1-NORMAL	(--)	BACK TO NORMAL not wanted for capture
C1-BPVL	(--)	BIPOLAR VIOLATION not wanted for capture
C1-CRCR	(--)	CRC ERROR not wanted for capture
C1-MERR	(--)	MULTIFRAME ERROR not wanted for capture
C1-FERR	(--)	FRAME ERROR not wanted for capture
C1-CEPTSPEC	(--)	CEPT SPECIAL ERROR not wanted for capture

8.2.3. FRAME LEVEL FILTER COMMANDS

C2=ALL	(--)	All frame types selected for capture
C2=NONE	(--)	No frames selected for capture
C2+SABME	(--)	SABME wanted for capture
C2+DISC	(--)	DISC wanted for capture
C2+UA	(--)	UA wanted for capture
C2+DM	(--)	DM wanted for capture
C2+FRMR	(--)	FRMR wanted for capture
C2+INV	(--)	Invalid frame wanted for capture
C2+I	(--)	I frame wanted for capture
C2+UI	(--)	UI wanted for capture
C2+RR	(--)	RR wanted for capture
C2+RNR	(--)	RNR wanted for capture
C2+REJ	(--)	REJ wanted for capture
C2+XID	(--)	XID wanted for capture
C2-SABME	(--)	SABME not wanted for capture
C2-DISC	(--)	DISC not wanted for capture
C2-UA	(--)	UA not wanted for capture
C2-DM	(--)	DM not wanted for capture
C2-FRMR	(--)	FRMR not wanted for capture
C2-INV	(--)	Invalid frame not wanted for capture
C2-I	(--)	I frame not wanted for capture
C2-UI	(--)	UI not wanted for capture
C2-RR	(--)	RR not wanted for capture
C2-RNR	(--)	RNR not wanted for capture
C2-REJ	(--)	REJ not wanted for capture
C2-XID	(--)	XID not wanted for capture
=CF_TEI	(TEI --)	Selects the TEI value which shall be captured.
<>CF_TEI	(TEI --)	Selects the TEI value which shall be ignored.
CF_TEI=ALL	(--)	Selects all TEI values for capture.
CF_TEI=127	(--)	Selects additionally TEI 127 for capture when another TEI value is already chosen (not 127). Otherwise there is no action.
CF_TEI<>127	(--)	Deselects TEI 127 if it was chosen by **CF_TEI=127**. Otherwise there is no action.

=CF_SAPI (SAPI --) Selects the SAPI value which shall be captured.

<>CF_SAPI (SAPI --) Selects the SAPI value which shall be ignored.

CF_SAPI=ALL (--) Selects all SAPI values for capture.

8.2.4. MESSAGE LEVEL FILTER COMMANDS

Messages are added or removed from the Capture filter list by initially entering the command M#ABCDEF, where ABCDEF represents the mnemonic string for a given message type. (For example, M#SETUP or M#SETUP_ACK). See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual part for a complete list of Layer 3 message identifiers. Thereafter the nominated message is added (**C+MSG**) or removed (**C-MSG**) from the Capture filter list. Note that adding a message that has already been added or removing one that has already been removed has no effect.

Invalid messages and undefined messages are specified using **M#INVALID** and **M#UNDEF** respectively. They are added or removed from the filter list in the same way as other messages.

Protocol discriminator and call reference values can be altered or added to the Capture filters with the following FORTH commands:

8.2.4.1. PROTOCOL DISCRIMINATOR

=F_P_DISC **(PD value --)**
Add the Protocol Discriminator event with the given PD value to the active filter list.

-F_P_DISC **(--)**
Remove the Protocol discriminator event from the active filter list.

8.2.4.2. CALL REFERENCE

=F_C_REF **(CR value --)**
Add the Call Reference event with the given CR value to the active filter list.

-F_C_REF **(--)**
Remove the Call Reference event from the active filter list.

For Example,

```
0x0F =F_P_DISC
4660 =F_C_REF
```

Note: The order in which filtering is performed is:
1. Protocol Discriminator, 2. Call Reference, 3. Message

8.3. DISK FILTERS

The Disk Filters will pass only selected frames to be saved to disk. Thus, the user may select specified frames and/or packet types to be saved on the disk and all others will be ignored.

The following commands allow selective frame type filtering before the data is passed to the Disk recording.

8.3.1. PHYSICAL LEVEL FILTER COMMANDS (BASIC ACCESS)

D1=ALL	(--)	All layer 1 events wanted for disk recording
D1=NONE	(--)	No layer 1 events wanted for disk recording
D1+ACTIVATED	(--)	ACTIVATION wanted for disk recording
D1+DEACTIVATED	(--)	DEACTIVATION wanted for disk recording
D1+LSTFRM	(--)	LOST FRAMING wanted for disk recording
D1+RECOVERY	(--)	RECOVERY wanted for disk recording
D1+RI2ERR	(--)	Too many RCV. ERROR wanted for disk recording
D1+UNDEFINED	(--)	CCITT UNDEFINED ERROR wanted for disk recording
D1-ACTIVATED	(--)	ACTIVATION not wanted for disk recording
D1-DEACTIVATED	(--)	DEACTIVATION not wanted for disk recording
D1-LSTFRM	(--)	LOST FRAMING not wanted for disk recording
D1-RECOVERY	(--)	RECOVERY not wanted for disk recording
D1-RI2ERR	(--)	Too many RCV ERROR not wanted for disk recording
D1-UNDEFINED	(--)	CCITT UNDEFINED ERROR not wanted for disk recording

8.3.2. PHYSICAL LEVEL FILTER COMMANDS (PRIMARY RATE ACCESS)

D1=ALL	(--)	All layer 1 events wanted for disk recording
D1=NONE	(--)	No layer 1 events wanted for disk recording
D1+RED_ALARM	(--)	RED ALARM wanted for disk recording
D1+YEL_ALARM	(--)	YELLOW ALARM wanted for disk recording
D1+LST_SIGL	(--)	LOST OF SIGNAL LEVEL wanted for disk recording
D1+LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK wanted for disk recording
D1+NORMAL	(--)	BACK TO NORMAL wanted for disk recording
D1+BPVL	(--)	BIPOLAR VIOLATION wanted for disk recording
D1+CRCR	(--)	CRC ERROR wanted for disk recording
D1+MERR	(--)	MULTIFRAME ERROR wanted for disk recording
D1+FERR	(--)	FRAME ERROR wanted for disk recording
D1+CEPTSPEC	(--)	CEPT SPECIAL ERROR wanted for disk recording
D1-RED_ALARM	(--)	RED ALARM not wanted for disk recording
D1-YEL_ALARM	(--)	YELLOW ALARM not wanted for disk recording
D1-LST_SIGL	(--)	LOST OF SIGNAL LEVEL not wanted for disk recording
D1-LST_PLOCK	(--)	LOST OF PHYSICAL LINK LOCK not wanted for disk recording
D1-NORMAL	(--)	BACK TO NORMAL not wanted for disk recording
D1-BPVL	(--)	BIPOLAR VIOLATION not wanted for disk recording
D1-CRCR	(--)	CRC ERROR not wanted for disk recording
D1-MERR	(--)	MULTIFRAME ERROR not wanted for disk recording
D1-FERR	(--)	FRAME ERROR not wanted for disk recording
D1-CEPTSPEC	(--)	CEPT SPECIAL ERROR not wanted for disk recording

8.3.3. FRAME LEVEL FILTER COMMANDS

D2=ALL	(--)	All frame types wanted for disk recording
D2=NONE	(--)	No frames wanted for disk recording
D2+SABME	(--)	SABME wanted for disk recording
D2+DISC	(--)	DISC wanted for disk recording
D2+UA	(--)	UA wanted for disk recording
D2+DM	(--)	DM wanted for disk recording
D2+FRMR	(--)	FRMR wanted for disk recording
D2+INV	(--)	Invalid frame wanted for disk recording
D2+I	(--)	I frame wanted for disk recording
D2+UI	(--)	UI wanted for disk recording
D2+RR	(--)	RR wanted for disk recording
D2+RNR	(--)	RNR wanted for disk recording
D2+REJ	(--)	REJ wanted for disk recording
D2+XID	(--)	XID wanted for disk recording
D2-SABME	(--)	SABME not wanted for disk recording
D2-DISC	(--)	DISC not wanted for disk recording
D2-UA	(--)	UA not wanted for disk recording
D2-DM	(--)	DM not wanted for disk recording
D2-FRMR	(--)	FRMR not wanted for disk recording
D2-INV	(--)	Invalid frame not wanted for disk recording
D2-I	(--)	I frame not wanted for disk recording
D2-UI	(--)	UI not wanted for disk recording
D2-RR	(--)	RR not wanted for disk recording
D2-RNR	(--)	RNR not wanted for disk recording
D2-REJ	(--)	REJ not wanted for disk recording
D2-XID	(--)	XID not wanted for disk recording
=DF_TEI	(TEI --)	Selects the TEI value which shall be recorded.
<>DF_TEI	(TEI --)	Selects the TEI value which shall be ignored.
DF_TEI=ALL	(--)	Selects all TEI values for disk recording.
DF_TEI=127	(--)	Selects additionally TEI 127 for disk recording when another TEI value is already chosen (not 127). Otherwise there is no action.
DF_TEI<>127	(--)	Deselects TEI 127 if it was chosen by **CF_TEI=127**. Otherwise there is no action.

=DF_SAPI (SAPI --) Selects the SAPI value which shall be recorded.

<>DF_SAPI (SAPI --) Selects the SAPI value which shall be ignored.

DF_SAPI=ALL (--) Selects all SAPI values for disk recording.

8.3.4. MESSAGE LEVEL FILTER COMMANDS

Messages are added or removed from the Disk filter list by initially entering the command M#ABCDEF, where ABCDEF represents the mnemonic string for a given message type. (For example, M#SETUP or M#SETUP_ACK). See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual part for a complete list of Layer 3 message identifiers. Thereafter the nominated message is added (**D+MSG**) or removed (**D-MSG**) from the Disk filter list. Note that adding a message that has already been added or removing one that has already been removed has no effect.

Invalid messages and undefined messages are specified using **M#INVALID** and **M#UNDEF** respectively. They are added or removed from the filter list in the same way as other messages.

Protocol discriminator and call reference values can be altered or added to the Disk filters with the following FORTH commands:

8.3.4.1. PROTOCOL DISCRIMINATOR

=F_P_DISC (PD value --)
Add the Protocol Discriminator event with the given PD value to the active filter list.

-F_P_DISC (--)
Remove the Protocol discriminator event from the active filter list.

8.3.4.2. CALL REFERENCE

=F_C_REF (CR value --)
Add the Call Reference event with the given CR value to the active filter list.

-F_C_REF (--)
Remove the Call Reference event from the active filter list.

For Example,

```
0x0F =F_P_DISC
4660 =F_C_REF
```

Note: The order in which filtering is performed is:
1. Protocol Discriminator, 2. Call Reference, 3. Message

9. PROGRAMMABLE TRIGGERS

The programmable triggers in the ISDN Monitor permit the user to select the events which are to be used to start, change or end a monitoring session. There are four event trigger machines (TR1, TR1, TR3, TR4). Event triggers relate to Layer 1, 2 or 3 events such as frames or messages.

The trigger actions (TA1, TA2, TA3, TA4) for each event trigger can be defined, such that they can include the definition of other triggers.

9.1. EVENT TRIGGERS

There are four independent event trigger machines in the ISDN Monitor. Each of these trigger machines can be set to trigger on any (frame) event, and execute the actions defined.

The event triggers are called TR1, TR2, TR3, and TR4 and they are processed in that order. The corresponding trigger actions (which are executed after the trigger event has occurred) are called TA1, TA2, TA3, and TA4.

Setup of the trigger events is accomplished by specifying the trigger, and adding or subtracting the protocol event from it.

Setup of the trigger actions is accomplished by using the MAKE procedure to define the action to be taken when a trigger event is found. Trigger actions consist of a sequence of Forth commands.

Triggers can be chained together for greater versatility. For example, it may be necessary to search all information frames for a particular string of characters. The first trigger could be used to find all information frames. Consequently, its trigger action would activate a second trigger that searches for the desired characters within the information frame. If the string was found, the trigger action defined for the second trigger would be executed.

9.2. COMMANDS TO SPECIFY TRIGGER EVENTS

The trigger machine is specified followed by the events to be added or deleted from this trigger. The format used is to specify the trigger, then the events.

9.2.1. TRIGGER MACHINE COMMANDS

TR1	(--)	Selects trigger machine TR1.
TR2	(--)	Selects trigger machine TR2.
TR3	(--)	Selects trigger machine TR3.
TR4	(--)	Selects trigger machine TR4.

9.2.2. TRIGGER EVENT COMMANDS

The trigger condition can be specified as anything (TR1 =ALL), nothing (TR1 =NONE), or a specific event or combination of events. The following sections list all the commands that can be used to specify the events.

9.2.2.1. TRIGGER DIRECTIONS

The source of the input stream upon which the triggers act can be specified using the following commands:

=USR	(--)	Trigger on events received from 'USER to NETWORK' direction.
=NET	(--)	Trigger on events received from 'NETWORK to USER' direction.
=BOTH	(--)	Trigger on events received from both directions.
=PLAYBACK	(--)	Trigger on events from disk or capture playback.

Eg. `TR1 =USR` or `TR1 =BOTH`

9.2.2.1.1. PHYSICAL LEVEL EVENTS (BASIC ACCESS)

The following list contains the FORTH commands which can be used to add "+" or remove "-" Layer 1 events from the active trigger list. Note: the "+/-" symbols indicate that either the "+" or the "-" signs (but not both) are to used as a prefix to the FORTH commands.

+/-ACTIVATE	(--)	Layer 1 activation event added/removed from the trigger list.
+/-DEACTIVATE	(--)	Layer 1 deactivation event added/removed from the trigger list.
+/-RI2ERR	(--)	Received **INFO 2** (in activated state) event added/removed from trigger list.
+/-LSTFRM	(--)	Lost Framing event added/removed from the trigger list.
+/-RECOVERY	(--)	Recovery (from previously reported error) event added/ removed the trigger list.
+/-UNDEFINED	(--)	Event not specified by CCITT added/removed from the trigger list.

9.2.2.1.2. PHYSICAL LEVEL EVENTS (PRIMARY RATE ACCESS)

+/-RED_ALARM	(--)	Red alarm added/removed from the trigger list.
+/-YEL_ALARM	(--)	Yellow alarm added/removed from the trigger list.
+/-LST_SIGL	(--)	Lost of signal level added/removed from the trigger list.
+/-LST_PLOCK	(--)	Lost of physical link lock added/removed from the trigger list.
+/-NORMAL	(--)	Layer 1 back to normal added/ removed the trigger list.
+/-BPVL	(--)	Bipolar Violation added/removed from the trigger list.
+/-CRCR	(--)	CRC error added/removed from the trigger list.
+/-MERR	(--)	Multiframe error added/removed from the trigger list.
+/-FERR	(--)	Frame error added/removed from the trigger list.
+/-CEPTSPEC	(--)	CEPT special error added/ removed the trigger list.

9.2.2.2. FRAME LEVEL EVENTS

The following list contains the FORTH commands which can be used to add "+" or remove "-" Layer 2 (frame) events from the active trigger list. Note, the "+/-" symbols indicate that either the "+" or the "-" signs (but not both) are to used as a prefix to the FORTH commands.

+/-SABME	(--)	**SABME** frame added/removed from the trigger list.
+/-DISC	(--)	**DISC** frame added/removed from the trigger list.
+/-UA	(--)	**UA** frame added/removed from the trigger list.
+/-DM	(--)	**DM** frame added/removed from the trigger list.
+/-I	(--)	**I** frame added/removed from the trigger list.
+/-UI	(--)	**UI** frame added/removed from the trigger list.
+/-RR	(--)	**RR** frame added/removed from the trigger list.
+/-RNR	(--)	**RNR** frame added/removed from the trigger list.
+/-REJ	(--)	**REJ** frame added/removed from the trigger list.
+/-FRMR	(--)	**FRMR** frame added/removed from the trigger list.
+/-XID	(--)	**XID** frame added/removed from the trigger list.
+/-INVFRM	(--)	**INVFRM** frame added/removed from the trigger list. The invalid frame one of a CRC error, ABORT, invalid address, or invalid control byte.

9.2.2.3. MESSAGE LEVEL EVENTS

Messages are added or removed from the active trigger by initially entering the command M#ABCDEF, where ABCDEF represents the mnemonic string for a given message type. (For example, **M#SETUP** or **M#SETUP_ACK**). See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual part for a complete list of Layer 3 message identifiers. Thereafter the nominated message can be added (**T+MSG**), or removed (**T-MSG**) from the active trigger list. Note that adding a message that has already been added or removing one that has already been removed has no effect.

Example: `M#SETUP T+MSG`

Invalid messages and undefined messages are specified using **M#INVALID** and **M#UNDEF** respectively. They are then added or removed from the trigger list in the same way as other messages.

9.2.2.4. MISCELLANEOUS EVENTS

=ALL	(--)	Add all the (protocol) events to the active trigger list.
=NONE	(--)	Remove all the (protocol) events from the active trigger list.
=STRING	**(string address --)**	Match trigger string. The string must be defined by quote pairs with " TEXT" for ASCII character strings or with X" 010BFF" for hex values.

Use the question mark symbol (? HEX 3F) to specify don't care positions within the character string. The string definition is started by the quote space pair and is ended by a quote. String definitions are limited to 64 characters long.

+/-STRING	(--)	The string trigger will be added "+" or removed "-" from the trigger events. The characters last defined using =STRING for the active trigger will be used for the string match comparison.
=TIME	**(year / month / day / hour / minute --)**	Event trigger at the selected time. The time must be entered in front of the =TIME in the format:

Year Month Day Hour Minute =TIME

The full year is entered (1989) and all other fields must be entered using the number value for them with a space separating each field. The time is entered in 24 hour format.

Example: `1989 1 1 15 5 TR1 =TIME`

This command will set the trigger 1 time to 1989 January 1, 15 hours (3 PM), 5 minutes.

Note: Do no type in leading zeros (eg. 08) as this will cause an error.

+/-TIME	(--)	The time trigger previously set with =TIME will be added "+" or removed "-" from the trigger events.
+/-DISK	(--)	Add/Remove the disk full event from the trigger list.
+/-CAPT	(--)	Add/Remove the Capture RAM full event from the trigger list.
=SAPI	(SAPI --)	Set the Service Access Point Idendification value (eg.16 =SAPI).
+/-SAPI	(--)	Add/Remove the SAPI event from the trigger list.
=TEI	(TEI --)	Set the Terminal End point Idendification value (eg. 64 =TEI).
+/-TEI	(--)	Add/Remove the TEI from the trigger list.
+/-DLCI	(--)	Add/Remove the Data Link Connection Identification event from trigger list. Note the DLCI is a logical AND of the SAPI and TEI commands.
=T_P_DISC	(PD value --)	SpecifY the value of the protocol discriminator and add it to the active trigger list.
-T_P_DISC	(--)	Remove the Protocol discriminator trigger from the active trigger list.
=T_C_REF	(CR value --)	Specifie the value of the call reference and add it to the active trigger list.
-T_C_REF	(--)	Remove the call reference trigger from the active trigger list.

9.2.3. EXAMPLES

`TR1 X" 010BFF" =STRING`	This command will start trigger 1 searching for the string with hex values 01, 0B, FF.
`TR1 =ALL`	Trigger 1 would be all events.
`TR1 -I`	I frames would be deleted from Trigger 1.
`TR2 =NONE`	Trigger 2 would be no events.
`TR2 +SABME`	SABME is added to trigger 2 events.
`TR1 =NONE +SABME +DISC`	would execute trigger 1 action (TA1) when a SABME, DISC is received.

9.3. DEFINING TRIGGER ACTIONS

Trigger actions are defined using the 'MAKE' word. The form used is:

```
MAKE  TA1  ENB_REC  BEEP ;
MAKE  TA2  REP_ON ;
MAKE  TA3  CAPT_ON ;
MAKE  TA4  DIS_REC  REP_OFF  CAPT   F ;
```

When the trigger event is found, the action that has been defined for this trigger will be executed. The actions can be any of the commands of the analysis package which or FORTH words that have been defined by the user with : and ;. In the example above, trigger 1 action has been defined as ENB_REC (Enable disk recording) and BEEP (beep the sonalert). The MAKE actions definition must always end with a semicolon (;).

HIGHLIGHT – The HIGHLIGHT command will cause the received report display for the trigger event to be highlighted. **HIGHLIGHT=RED** and **HIGHLIGHT=BLUE** would select red or blue background color for highlighting reports.

INDEX

Symbols

+ACTIVATE, 9-2
+BPVL, 9-3
+CAPT, 9-5
+CEPTSPEC, 9-3
+CRCR, 9-3
+DEACTICATE, 9-2
+DISC, 9-3
+DISK, 9-5
+DLCI, 9-5
+DM, 9-3
+FERR, 9-3
+FRMR, 9-3
+I, 9-3
+INVFRM, 9-3
+LST_PLOCK, 9-3
+LST_SIGL, 9-3
+LSTFRM, 9-2
+MERR, 9-3
+NORMAL, 9-3
+RECOVERY, 9-2
+RED_ALARM, 9-3
+REJ, 9-3
+RI2ERR, 9-2
+RNR, 9-3
+RR, 9-3
+SABME, 9-3
+SAPI, 9-5
+STRING, 9-4
+TEI, 9-5
+TIME, 9-5
+UA, 9-3
+UI, 9-3
+UNDEFINED, 9-2
+XID, 9-3
+YEL_ALARM, 9-3
-ACTIVATE, 9-2
-BPVL, 9-3
-CAPT, 9-5
-CEPTSPEC, 9-3
-CRCR, 9-3
-DEACTIVATE, 9-2
-DISC, 9-3
-DISK, 9-5
-DLCI, 9-5
-DM, 9-3
-F_C_REF, 8-8, 8-12
-F_P_DISC, 8-4, 8-8, 8-12
-FERR, 9-3
-FRMR, 9-3
-I, 9-3
-INVFRM, 9-3
-LST_PLOCK, 9-3
-LST_SIGL, 9-3
-LSTFRM, 9-2
-MERR, 9-3
-NORMAL, 9-3
-RECOVERY, 9-2
-RED_ALARM, 9-3
-REJ, 9-3
-RI2ERR, 9-2
-RNR, 9-3
-RR, 9-3
-SABME, 9-3
-SAPI, 9-5
-STRING, 9-4
-T_C_REF, 9-5
-T_P_DISC, 9-5
-TEI, 9-5
-TIME, 9-5
-UA, 9-3
-UI, 9-3
-UNDEFINED, 9-2
-XID, 9-3
-YEL-ALARM, 9-3
=ALL, 9-4
=BOTH, 9-2
=CF_SAPI, 8-8
=CF_TEI, 8-7
=DF_SAPI, 8-12
=DF_TEI, 8-11
=F_C_REF, 8-4, 8-8, 8-12
=F_P_DISC, 8-4, 8-8, 8-12
=NET, 9-2

=NONE, 9-4
=PLAYBACK, 9-2
=RF_SAPI, 8-4
=RF_TEI, 8-3
=SAPI, 9-5
=STRING, 9-4
=T_C_REF, 9-5
=T_P_DISC, 9-5
=TEI, 9-5
=TIME, 9-4
=TITLE, 6-3
=TSLOT, 7-3
=USR, 9-2

B

BACKWARD, 7-2
BLOCK_ON, 5-10
BOTTOM, 7-2

C

C/I_OFF, 5-9
C/I_ON, 5-9
C1+ACTIVATED, 8-5
C1+BPVL, 8-6
C1+CEPTSPEC, 8-6
C1+CRCR, 8-6
C1+DEACTIVATED, 8-5
C1+FERR, 8-6
C1+LST_PLOCK, 8-6
C1+LST_SIGL, 8-6
C1+LSTFRM, 8-5
C1+MERR, 8-6
C1+NORMAL, 8-6
C1+RECOVERY, 8-5
C1+RED_ALARM, 8-6
C1+RI2ERR, 8-5
C1+UNDEFINED, 8-5
C1+YEL_ALARM, 8-6
C1-ACTIVATED, 8-5
C1-BPVL, 8-6
C1-CEPTSPEC, 8-6
C1-CRCR, 8-6
C1-DEACITVATED, 8-5
C1-FERR, 8-6
C1-LST_PLOCK, 8-6
C1-LST_SIGL, 8-6
C1-LSTFRM, 8-5
C1-MERR, 8-6
C1-NORMAL, 8-6
C1-RECOVERY, 8-5
C1-RED_ALARM, 8-6
C1-RI2ERR, 8-5
C1-YEL-ALARM, 8-6
C1=ALL, 8-5, 8-6
C1=NONE, 8-5, 8-6
C1_UNDEFINED, 8-5
C2+DISC, 8-7
C2+DM, 8-7
C2+FRMR, 8-7
C2+I, 8-7
C2+INV, 8-7
C2+REJ, 8-7
C2+RNR, 8-7
C2+RR, 8-7
C2+SABME, 8-7
C2+UA, 8-7
C2+UI, 8-7
C2+XID, 8-7
C2-DISC, 8-7
C2-DM, 8-7
C2-FRMR, 8-7
C2-I, 8-7
C2-INV, 8-7
C2-REJ, 8-7
C2-RNR, 8-7
C2-RR, 8-7
C2-SABME, 8-7
C2-UA, 8-7
C2-UI, 8-7
C2-XID, 8-7
C2=ALL, 8-7
C2=NONE, 8-7
CAPT_FULL, 7-1
CAPT_OFF, 7-1

CAPT_ON, 7-1
CAPT_WRAP, 7-1
CF_SAPI, 8-8
CF_SAPI=ALL, 8-8
CF_TEI, 8-7
CF_TEI=127, 8-7
CF_TEI=ALL, 8-7
CF_TEI127, 8-7
CLEAR_CRT, 7-1
COMMAND, 4-2
CS=ASCII, 5-10
CS=EBCDIC, 5-10
CS=HEX, 5-10
CTOD_OFF, 7-2
CTOD_ON, 7-2

D

D1+ACTIVATED, 8-9
D1+BPVL, 8-10
D1+CEPTSPEC, 8-10
D1+CRCR, 8-10
D1+DEACTIVATED, 8-9
D1+FERR, 8-10
D1+LST_PLOCK, 8-10
D1+LST_SIGL, 8-10
D1+LSTFRM, 8-9
D1+MERR, 8-10
D1+NORMAL, 8-10
D1+RECOVERY, 8-9
D1+RED_ALARM, 8-10
D1+RI2ERR, 8-9
D1+UNDEFINED, 8-9
D1+YEL_ALARM, 8-10
D1-ACTIVATED, 8-9
D1-BPVL, 8-10
D1-CEPTSPEC, 8-10
D1-CRCR, 8-10
D1-DEACTIVATED, 8-9
D1-FERR, 8-10
D1-LST_PLOCK, 8-10
D1-LST_SIGL, 8-10
D1-LSTFRM, 8-9
D1-MERR, 8-10
D1-NORMAL, 8-10
D1-RECOVERY, 8-9
D1-RED_ALARM, 8-10
D1-RI2ERR, 8-9
D1-UNDEFINED, 8-9
D1-YEL_ALARM, 8-10
D1=ALL, 8-9, 8-10
D1=NONE, 8-9, 8-10
D2+DISC, 8-11
D2+DM, 8-11
D2+FRMR, 8-11
D2+I, 8-11
D2+INV, 8-11
D2+REJ, 8-11
D2+RNR, 8-11
D2+RR, 8-11
D2+SABME, 8-11
D2+UA, 8-11
D2+XID, 8-11
D2-DISC, 8-11
D2-DM, 8-11
D2-FRMR, 8-11
D2-I, 8-11
D2-INV, 8-11
D2-REJ, 8-11
D2-RNR, 8-11
D2-RR, 8-11
D2-SABME, 8-11
D2-UA, 8-11
D2-UI, 8-11
D2-XID, 8-11
D2=ALL, 8-11
D2=NONE, 8-11
D2_UI, 8-11
DF_DAPI=ALL, 8-12
DF_SAPI, 8-12
DF_TEI, 8-11
DF_TEI=127, 8-11
DF_TEI=ALL, 8-11
DF_TEI127, 8-11
DIS_REC, 6-3

DISK_OFF, 6-3
DR0, 7-1
DR1, 7-1

E

ENB_REC, 6-3
EXT_BCH_OFF, 7-3
EXT_BCH_ON, 7-3

F

FAST_MODE, 6-3
FFMT=CRC4, 7-3
FFMT=D4, 7-3
FFMT=ESF, 7-3
FFMT=PCM, 7-3
FORWARD, 7-2
FREEZE, 4-2, 7-1
FRM_CHAR, 5-9
FRM_COMP, 5-9
FRM_HEX, 5-9
FRM_OFF, 5-9
FRM_SHRT, 5-9
FROM_CAPT, 6-3
FROM_DISK, 6-3

H

HALT, 4-2, 7-1
HELP, 4-2
HIGHLIGHT, 9-6

I

I/F=V.11, 7-4
I/F=V.28, 7-4
I/F=V.35, 7-4
I/F=V.36, 7-4
IHDLC_OFF, 7-3
IHDLC_ON, 7-3
IMPED=120, 7-3
IMPED=75, 7-3
IMPED=HI, 7-3

L

L1_COMP, 5-9
L1_OFF, 5-9
L1_SHRT, 5-9
LCOD=AMI, 7-3
LCOD=B8ZS, 7-3
LCOD=HDB3, 7-3
LENGTH=0, 7-3
LENGTH=2, 7-3

M

MENU, 4-2
MON_KEYS, 4-2
MONITOR, 4-2, 7-1
MSG_CHAR, 5-9
MSG_COMP, 5-9
MSG_HEX, 5-9
MSG_MED1, 5-9
MSG_MED2, 5-9
MSG_MNEM, 5-9
MSG_OFF, 5-9
MSG_SHRT, 5-9

N

NORM_MODE, 6-3

P

PLAY=OFF, 6-2
PLAY=ON, 6-2
PLAY_1, 6-2
PLAY_2, 6-2
PLAY_3, 6-2
PLAY_4, 6-2
PLAY_7, 6-2
PLAY_8, 6-2
PLAYBACK, 6-3
PRINT_OFF, 7-2
PRINT_ON, 7-2

Q

QUIT_COM, 4-2

QUIT_HELP, 4-2
QUIT_MENU, 4-2
QUIT_TRA, 7-2

R

R1+ACTIVATED, 8-1
R1+BPVL, 8-2
R1+CEPTSPEC, 8-2
R1+CRCR, 8-2
R1+DEACTIVATED, 8-1
R1+FERR, 8-2
R1+LST_PLOCK, 8-2
R1+LST_SIGL, 8-2
R1+LSTFRM, 8-1
R1+MERR, 8-2
R1+NORMAL, 8-2
R1+RECOVERY, 8-1
R1+RED_ALARM, 8-2
R1+RI2ERR, 8-1
R1+UNDEFINED, 8-1
R1+YEL_ALARM, 8-2
R1-ACTIVATED, 8-1
R1-CEPTSPEC, 8-2
R1-CRCR, 8-2
R1-DEACTIVATED, 8-1
R1-FERR, 8-2
R1-LST_PLOCK, 8-2
R1-LST_SEGL, 8-2
R1-LSTFRM, 8-1
R1-MERR, 8-2
R1-RECOVERY, 8-1
R1-RED_ALARM, 8-2
R1-RI2ERR, 8-1
R1-UNDEFINED, 8-1
R1-YEL_ALARM, 8-2
R1=ALL, 8-1, 8-2
R1=NONE, 8-1, 8-2
R1_BPVL, 8-2
R1_NORMAL, 8-2
R2+DISC, 8-3
R2+DM, 8-3
R2+FRMR, 8-3
R2+I, 8-3
R2+INV, 8-3
R2+REJ, 8-3
R2+RNR, 8-3
R2+RR, 8-3
R2+SABME, 8-3
R2+UA, 8-3
R2+UI, 8-3
R2+XID, 8-3
R2-DISC, 8-3
R2-DM, 8-3
R2-FRMR, 8-3
R2-I, 8-3
R2-INV, 8-3
R2-REJ, 8-3
R2-RNR, 8-3
R2-RR, 8-3
R2-SABME, 8-3
R2-UA, 8-3
R2-UI, 8-3
R2-XID, 8-3
R2=ALL, 8-3
R2=NONE, 8-3
RECORD, 6-3
REP_OFF, 5-9
REP_ON, 5-9
RF_SAPI, 8-4
RF_SAPI=ALL, 8-4
RF_TEI, 8-3
RF_TEI=127, 8-3
RF_TEI=ALL, 8-3
RF_TEI127, 8-3

S

SCRN_BACK, 7-2
SCRN_FWD, 7-2
SHUTDOWN_OFF, 5-10
SHUTDOWN_ON, 5-10
SPLIT_OFF, 5-10
SPLIT_ON, 5-10

T

T+MSG, 9-4

T-MSG, 9-4
TA1, 9-6
TA2, 9-6
TA3, 9-6
TA4, 9-6
TIME_HOUR, 5-10
TIME_ON, 5-10
TIME_SEC, 5-10
TIME_YEAR, 5-10
TM_KEYS, 4-2
TOP, 7-2
TR1, 9-2
TR2, 9-2
TR3, 9-2
TR4, 9-2
TRA_ALL, 7-2
TRA_END, 7-2
TRA_START, 7-2
TRANSFER, 7-2

W

WD0, 7-1

ISDN EMULATION / SIMULATION

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version : V3.0
System Version : MF 2.2

Revision Date : January 1991

1. INTRODUCTION 1-1

2. MENUS 2-1

2.1. SIMULATION MENU 2-1
2.1.1. LAYER 1 SIMULATION MENU (primary rate interface) 2-2
2.1.1.1. ERROR GENERATION MENU (primary rate interface) 2-3
2.1.2. LAYER 1 SIMULATION MENU (basic access interface) 2-4
2.1.3. LAYER 2 SIMULATION MENU 2-5
2.1.4. LAYER 2 INITIALISATION MENU 2-6
2.1.5. TEI MANAGEMENT MENU 2-7
2.1.6. LAYER 3 MENUS 2-7

2.2. STATUS MENU 2-8

3. MESSAGE BUILDING SYSTEM 3-1

3.1. INTRODUCTION 3-1

3.2. ISDN MESSAGE STRUCTURE 3-1

3.3. THE MENU BUILDING METHOD: AN OVERVIEW 3-3
3.3.1. Message Building 3-6
3.3.2. Message Display 3-7
3.3.3. Message Editing 3-7
3.3.4. Message Storage 3-7

3.4. THE MESSAGE BUILDING SYSTEM SETUP MENU 3-8
3.4.1. Changing the Call Reference Value 3-8

3.5. THE MESSAGE SELECTION MENU 3-9
3.5.1. Generating a Message in the manual Mode 3-10
3.5.1.1. Selecting a Message Type 3-10
3.5.2. Generating a Message in the automatic Mode 3-10

3.6. THE INFORMATION ELEMENT SELECTION MENU 3-11
3.6.1. Building Information Elements (manual mode) 3-13
3.6.2. Selecting Information Elements (auto mode) 3-13
3.6.3. Altering the Information Element Octets 3-13
3.6.4. Return to the Message Selection Menu 3-13

3.7. THE OCTET SELECTION MENU 3-14
3.7.1. Modification of Octets and Fields into the Selected IE 3-14

3.8. THE HEXEDITOR MENU 3-15
3.8.1. Editing the contents of the Editor Buffer 3-15
3.8.2. Leaving the hexeditor 3-16

3.9. THE MESSAGE POOL MANAGEMENT 3-16

4. LAYER 2 EMULATION 4-1

4.1. SEND COMMANDS 4-1
4.1.1. COMMANDS EXECUTED BY INVOKING THE STATE MACHINE . 4-1
4.1.1.1. LINK LAYER STATE MACHINE 4-1
4.1.1.2. MANAGEMENT LAYER STATE MACHINE 4-2
4.1.2. COMMANDS EXECUTED OUTSIDE THE CONTROL OF THE STATE MACHINE 4-3

4.2. SERVICE PRIMITIVES 4-5

4.3. SETTING VARIABLES 4-7
4.3.1. DATA LINK LAYER CONTROLLING VARIABLES 4-7
4.3.2. DATA BUFFER DEFINITIONS 4-13

4.4. MISCELLANEOUS 4-14
4.4.1. EMULATION MODES 4-14
4.4.2. EMULATION CONTROL 4-14
4.4.3. TRACING OF EMULATION EVENTS 4-15
4.4.4. COMMANDS FOR PRIMARY RATE ACCESS 4-16
4.4.5. COMMANDS FOR BASIC ACCESS 4-17
4.4.6. COMMANDS FOR V/X INTERFACE 4-18

4.5. LIST OF COMMUNICATION VARIABLES 4-19
4.5.1. TRANSMIT VARIABLES 4-19
4.5.2. RECEIVE VARIABLES 4-20
4.5.2.1. FRAME# EVENT VARIABLES 4-20
4.5.2.2. LAYER1# EVENT VARIABLES 4-23
4.5.2.3. FUNCTION-KEY# , STRING-ENTRY# EVENTS 4-24
4.5.2.4. TIME-OUT# , L1-TIME-OUT# AND TM-TIME-OUT# EVENTS . 4-24
4.5.2.5. TM-IPC# EVENTS 4-24

5. MISCELLANEOUS NOTES 5-1

5.1. "OUT OF BUFFERS" MESSAGE 5-1

5.2. XID PROCEDURES 5-1

5.3. DISABLING THE XID PROCEDURES 5-3

5.4. OVERLOADING TRANSMIT & RECEIVE BUFFERS 5-3

5.5. THE USE OF FRMR FRAMES 5-3

5.6. TEI MANAGEMENT PROCEDURES 5-4

6. LAYER 3 SIMULATION 6-1

6.1. INTRODUCTION 6-1

6.2. FUNCTIONAL GROUPS 6-2

6.3. TEST MANAGER SUPPORT FUNCTIONS 6-4

6.3.1. Main FORTH Dictionary 6-4
6.3.2. Layer 2 Emulation 6-4
6.3.3. Layer 3 Decoder 6-5
6.3.4. ISDN Monitor 6-5
6.3.5. Message Building System 6-5
6.3.6. Layer 3 Connection Management 6-7
6.3.6.1. Connection-related Data Management 6-9
6.3.6.2. Call Reference Management 6-10
6.3.6.3. Information Element Management 6-11
6.3.6.4. B-Channel Management 6-12
6.3.6.5. Timer Management 6-13
6.3.6.6. Connection Multiplexer 6-14
6.3.7. B / D Channel – Communication 6-18
6.3.7.1. Introduction 6-18
6.3.7.2. Description of Service Primitives 6-18
6.3.7.3. Communication Strategy 6-19

6.4. TEST MANAGER 6-20
6.4.1. Introduction 6-20
6.4.2. Evaluation of Layer 3 Events 6-20
6.4.3. Sending Messages 6-21

6.5. LAYER 3 MANAGEMENT USER INTERFACE 6-22
6.5.1. Connection Management 6-22
6.5.2. Connection Parameter Access 6-23
6.5.3. Call Reference Management 6-24
6.5.4. Information Element Buffer and Parameter Management 6-25
6.5.5. B-Channel Management 6-26
6.5.6. Miscellaneous 6-27

6.6. MESSAGE BUFFER POOL USER INTERFACE 6-28

6.7. PROGRAMMING METHOD OF GENERATING MESSAGES 6-33
6.7.1. Message Generation Commands 6-33
6.7.2. Automatic Message Generation Mode 6-34
6.7.3. Information Element Generation Commands 6-35

APPENDIX A-1

A.1. STATES FOR THE LAYER 2, THE ASP AND THE ML A-1

A.2. PRIMITIVE AND CONSTANT NAMES A-2

INDEX I-1

1. INTRODUCTION

This guide is intended to provide all necessary information for the organization of the ISDN Software package, independent from the selected interface (Basic Access, Primary Rate or V/X). The ISDN Software can emulate the functions of either the User Layer 2 protocol or the Network Layer 2 protocol (including full TEI assignment procedures). The features provided from within the Layer 2 Reference Emulation portion of the ISDN Test Pak are a complete set of the "tools" used in the development of the protocol implementation.

Sections 2 and 3 of this document describe the various capabilities of the menu system. Section 2 concentrates on those functions relating to the test set environment and the control of the layer 2 software, whereas Section 3 provides details of the layer 3 message building system.

Section 4 lists numerous commands that exist to provide an enhanced user interface to the Layer 2 emulation. Two basic command types are available to permit interaction from both within and without the Layer 2 state machine. Other command types exist to directly control the values of many internal variables to modify the behavior of the state machine. In general, these commands are used to invoke Layer 2 functions directly from the keyboard, or alternatively, through automatic Test Manager test sequences (see Test Manager User Manual). Section 4 also lists most of the emulation variables (useful in Testmanager environments).

Currently the Layer 2 Emulation supports 8 multiple TEI assignments (it is possible to handle either 8 data links in NETWORK Emulation or to simulate 8 Terminal Equipments). Therefore most of the commands described in the following sections act on the currently selected data link (e.g. the selected Connection Endpoint Suffix CES). Selecting a particular CES can be done via the menu system (see Section 2.1.4) or, if in a Testmanager environment, with some of the Service Primitives described in Section 4.

Section 5 is a collection of application designer notes providing a more detailed discussion of certain protocol and hardware aspects.

Section 6 describes the environment for a simulation of the Layer 3 (Network Layer) which is based on the testmanager and the message building system.

Appendix A lists the states of the L.2, ML and ASP state machines as well as the LAPD primitives.

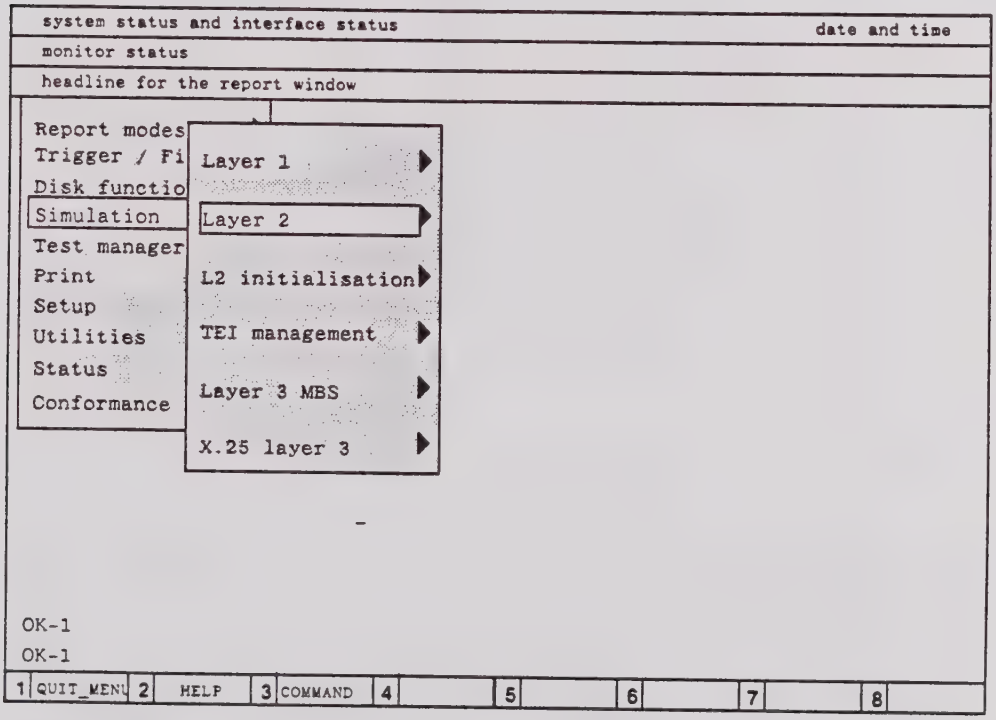
IMPLEMENTATION STANDARDS

The Layer 2 is implemented in accordance with the CCITT Blue Book Recommendations Q.920 and Q.921.

2. MENUS

The menu system of the K1197 and the menus of the ISDN monitor software are also available for the ISDN emulation/simulation software. Refer to the ISDN MONITOR USER MANUAL for detailed information about the menu system and these menus. Some menus which are not available in the monitor software are explained below.

2.1. SIMULATION MENU

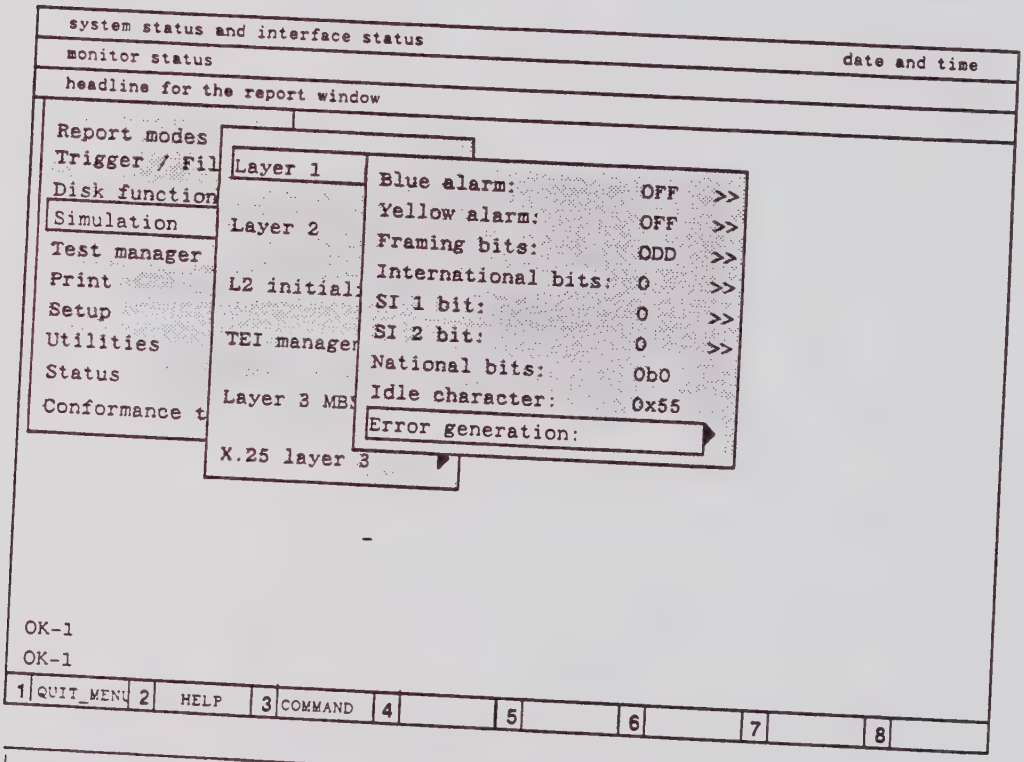


MENU SELECTION:

Layer 1	: Enter layer 1 simulation menu
Layer 2	: Enter layer 2 simulation menu
L2 initialisation	: Enter menu for layer 2 initialisation
TEI management	: Enter menu for TEI management functions
Layer 3 MBS	: Enter layer 3 message building system
X.25 layer 3	: Enter menu for X.25 data in layer 3

2.1.1. LAYER 1 SIMULATION MENU (primary rate interface)

The following menu is only available for primary rate interface. It is possible to set particular bits and alarms on the layer 1.

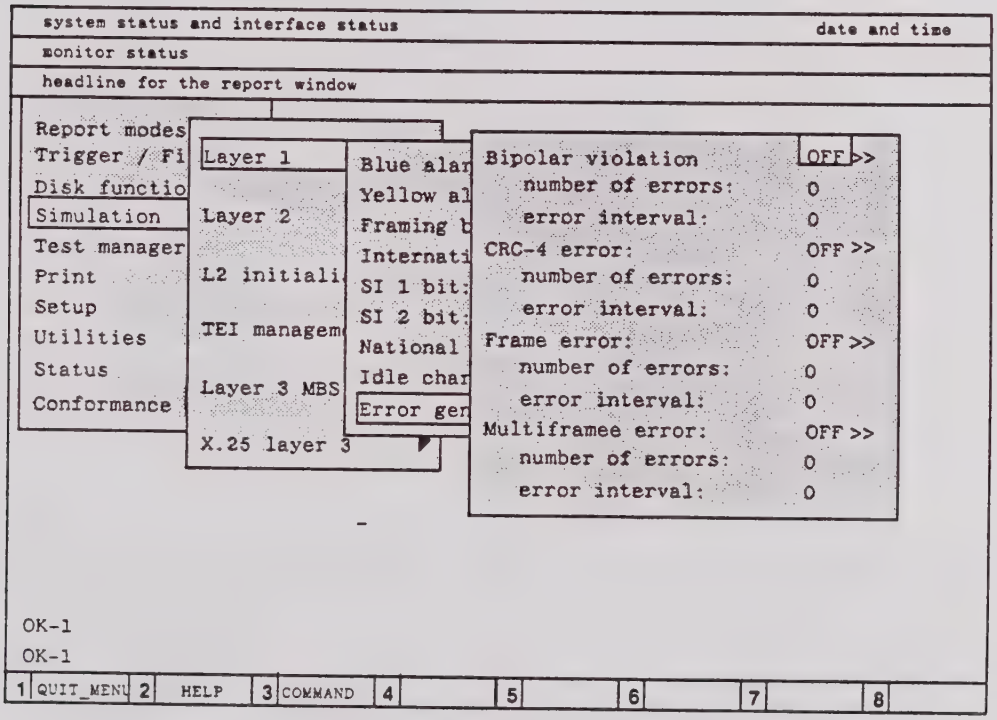


MENU SELECTION:

Blue alarm	: Select generation of BLUE alarm
Yellow alarm	: Select generation of YELLOW alarm
Framing bits	: Select location of framing bits
International bits	: Select international bits
SI 1 bit	: Select SI 1 bit
SI 2 bit	: Select SI 2 bit
National bits	: Set national bits
Idle character	: Set idle character
Error generation	: Enter menu for error generation on layer 1

2.1.1.1. ERROR GENERATION MENU (primary rate interface)

The following menu is only available for primary rate interface. It is possible to generate different errors on the layer 1. For the purpose of activating the error generator, the user first needs to specify how many invalid frames in a total rate of multiframes should be sent for each error type. The total rate (error interval) must also be specified (and must be greater than 40 multiframes). The errors are generated such that in every interval first the number of invalid frames will be generated and the rest of frames in the interval will be correct.

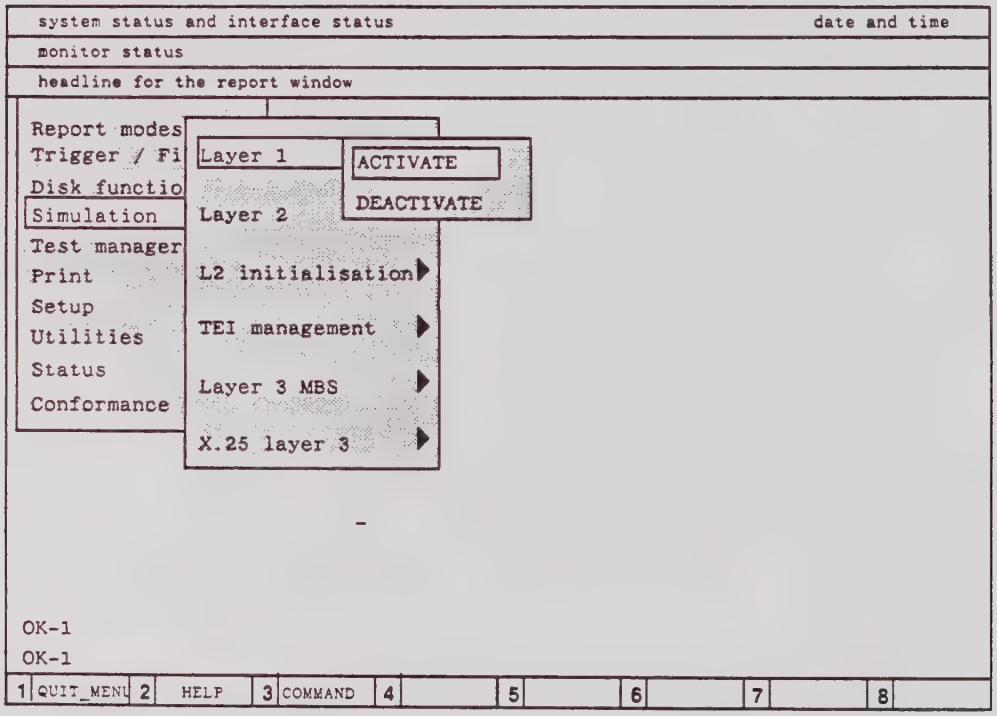


MENU SELECTION:

Bipolar violation	: Select generation of bipolar violation errors
CRC-4 error	: Select generation of CRC-4 errors
Frame error	: Select generation of frame errors
Multiframe error	: Select generation of multiframe errors

2.1.2. LAYER 1 SIMULATION MENU (basic access interface)

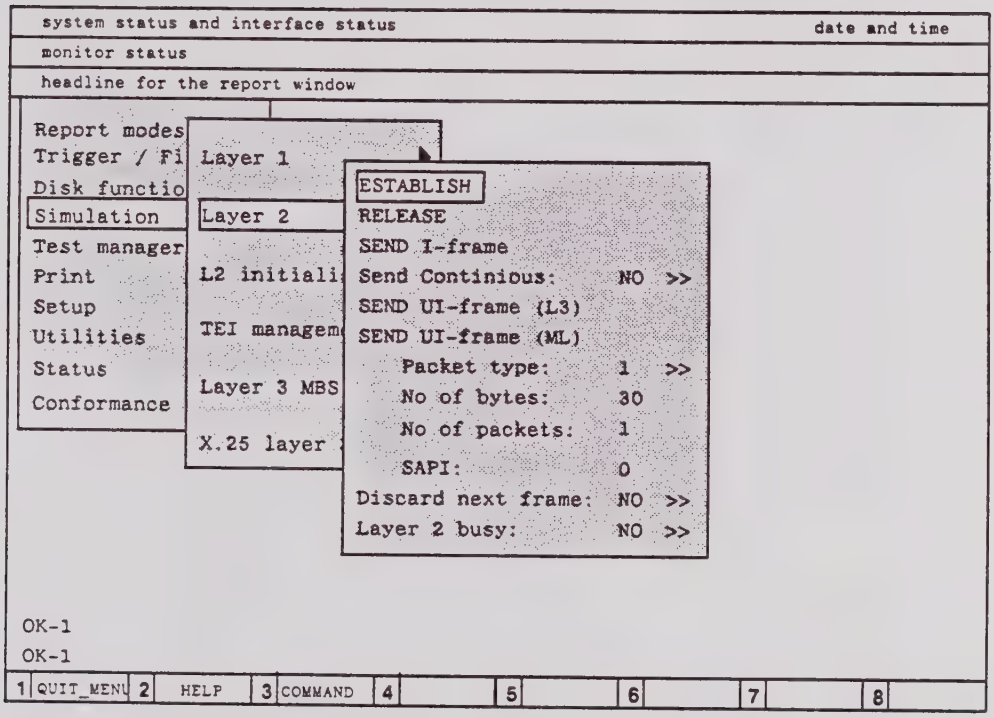
The following menu is only available for basic access interface.



MENU SELECTION:

ACTIVATE	: Activate the layer 1 (S-bus)
DEACTIVATE	: Deactivate the layer 1 (S-bus) (NT emulation only)
ERROR RESPONSE	: PH-ERROR-RESPONSE on layer 1 (TE emulation only)

2.1.3. LAYER 2 SIMULATION MENU

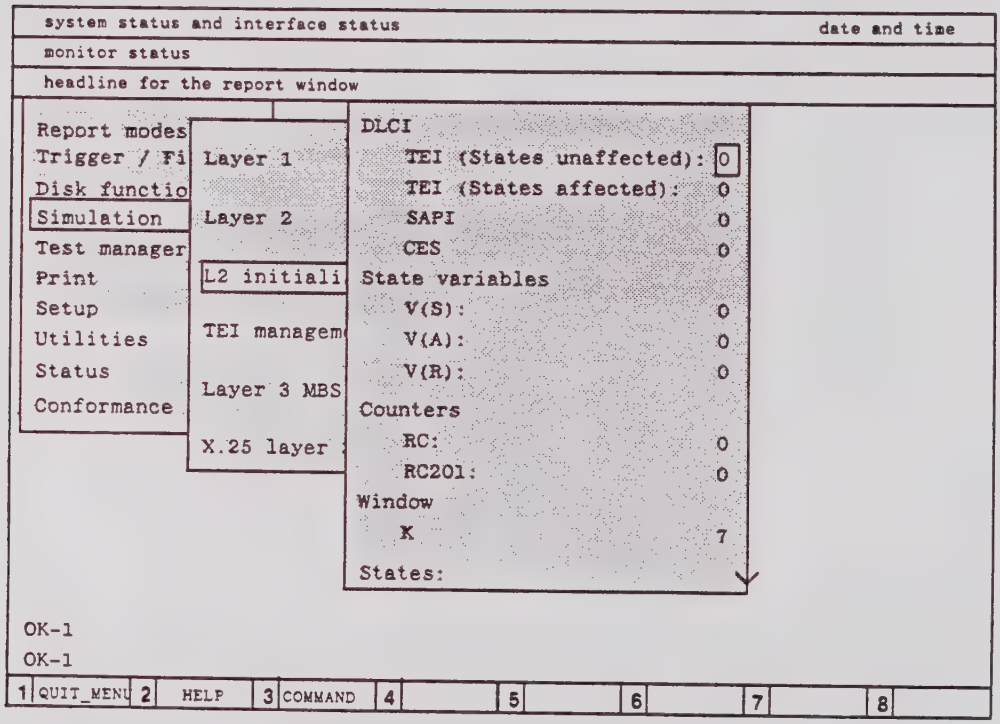


MENU SELECTION:

ESTABLISH	: Establish a data link connection on layer 2
RELEASE	: Transmit a DISC frame to release the current connection
SEND I-frame	: Transmit an I-frame
Send continious	: Select continuous transmission of I-frames
SEND UI-frame (L3)	: Transmit a UI-frame originating from the layer 3
SEND UI-frame (ML)	: Transmit a UI-frame originating from the management layer
Packet type	: Select predefined packet type for transmission 1 : sequentially number data bytes 2 : all 1's data bytes 3 : all 0's data bytes 4 : data bytes with alternating 1 & 0 bit pattern
No of bytes	: Set number of data bytes sent in each I-frame
No of packets	: Set the total number of I-frames to be transmitted
SAPI	: Set the SAPI to be transmitted with each I-frame
Discard next frame	: Select if the next frame received by the layer 2 should be discarded
Layer 2 busy	: Select if the layer 2 should response by transmitting an RR frame (L2 not busy) or an RNR-frame (L2 busy) to the peer layer

2.1.4. LAYER 2 INITIALISATION MENU

The following menu is not completely displayed on the screen, but can be scrolled by the cursor up and cursor down function keys. It gives the ability to initialize the TE (terminal equipment) / data link variables and others.

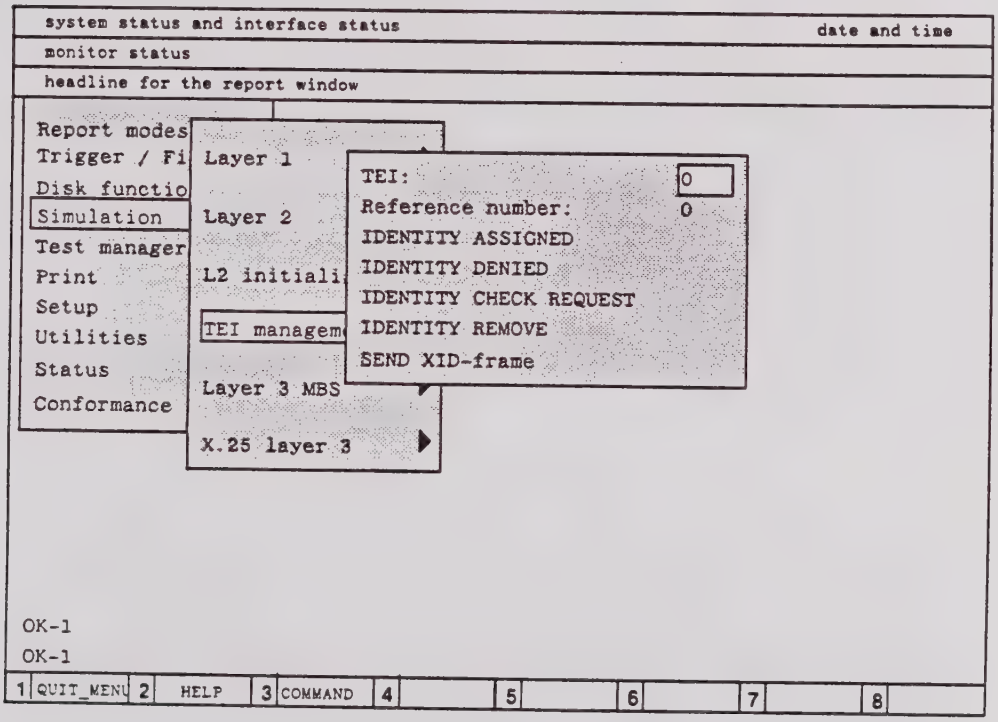


MENU SELECTION:

TEI (states unaffected)	: Set TEI (0-126), where the states are not affected
TEI (states affected)	: Set TEI (0-126), where the layer 2 state is changed to state 4
SAPI	: Set SAPI (0-63)
CES	: Set current CES (i.e. the link to be used) (0-7)
V(S)	: Set send state variable
V(A)	: Set acknowledge state variable
V(R)	: Set receive state variable
RC	: Set retransmission counter
RC201	: Set retransmission counter 201 (NT emulation only)
K	: Set window size for numbered information frames (current TE)
L2 state	: Set the layer 2 state of the current TE
ASP state	: Set the state of the NETWORK assignment source point (ASP)
ML state	: Set the state of the USER management layer (ML)
UI flag	: Set the indication of a UI frame with transmission pending (relevant in layer 2 states 2 and 3 only)
T203	: Set the layer 2 polling action on expiry of the T203 idle link timer (10 sec. as default value)

2.1.5. TEI MANAGEMENT MENU

The following menu allows to send commands of the management layer state machine by invoking the state machine.



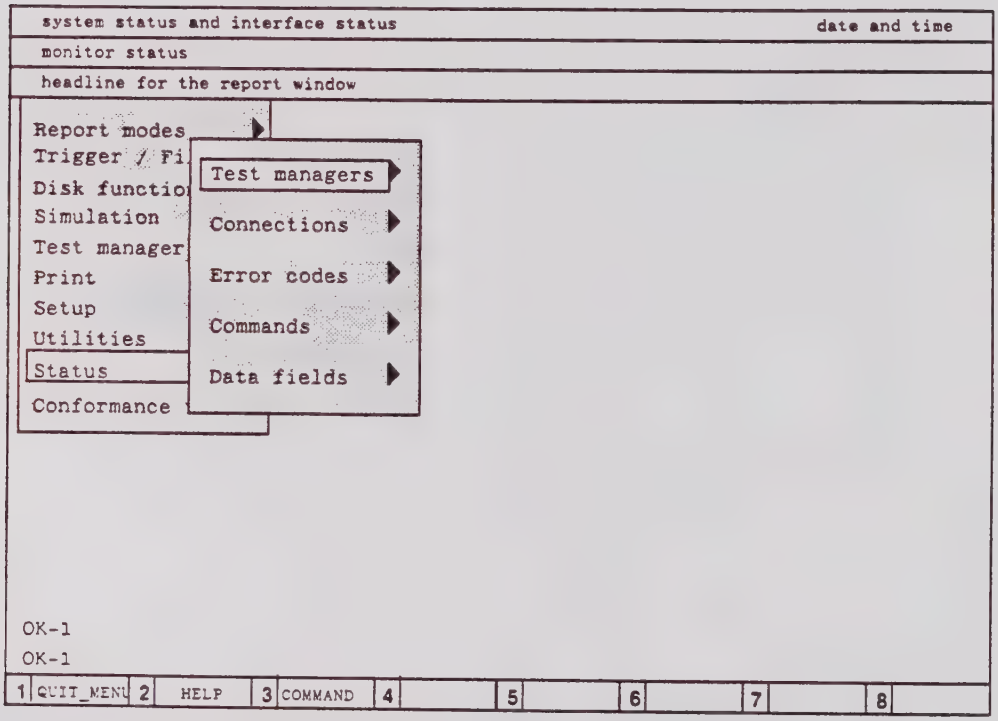
MENU SELECTION:

TEI	: Set TEI to be used in TEI management commands
Reference number	: Set reference number used in management commands
IDENTITY ASSIGNED	: Send "identity assigned" frame (NT emulation only)
IDENTITY DENIED	: Send "identity denied" frame (NT emulation only)
IDENTITY CHECK REQUEST	: Send "identity check request" frame (NT emulation only)
IDENTITY REMOVE	: Send "identity remove" frame (NT emulation only)
IDENTITY REQUEST	: Send "identity request" frame (TE emulation only)
IDENTITY CHECK RESPONSE	: Send "identity check response" frame (TE emulation only)
IDENTITY VERIFY	: Send "identity verify" frame (TE emulation only)
SEND XID-frame	: Send XID command frame

2.1.6. LAYER 3 MENUS

Refer to chapter 3 for detailled information about the layer 3 message building system and to X.25 LAYER 3 D-CHANNEL MANUAL for information about X.25 data in layer 3.

2.2. STATUS MENU



MENU SELECTION:

Test managers	: Display overview of test manager states
Connections	: Display overview of all assigned data links and variables
Error codes	: Display amount of MDL error codes
Commands	: Display a list of commands to change layer 2 variables
Data fields	: Display contents of the 8 data field defined by MAKE_DATA

3. MESSAGE BUILDING SYSTEM

3.1. INTRODUCTION

ISDN Layer 3 messages are complex. It is therefore desirable that users be given assistance to create them. The K1197 provides a Message Building System for this purpose. Such user generated messages (whether valid or deliberately invalid) can be used to test the response of ISDN equipment.

Two distinct methods for generating messages are available in the Message Building System. The Menu Building Method is a menu driven message building, display, editing and storing utility. A great deal of information is presented to the user as a series of menus, from which selections are made in order to create a message. This method is particularly useful to the novice ISDN user.

Messages can also be created using FORTH commands in the Program Building Method. This method is particularly suited to the experienced ISDN user who also has some knowledge of the FORTH programming language.

3.2. ISDN MESSAGE STRUCTURE

ISDN Layer 3 messages are made up of information elements, the information elements consist of one or more octets (8 bit bytes) and each octet contains one or more fields or parameters.

Messages are used by the Layer 3 protocol to convey information to set-up, change or disconnect a wide range of communication services. The message names relate to the function of the message. Each message contains information needed to perform its function. Some of the information is mandatory in every message, some is mandatory for particular messages and some is optional--that is, can be used in particular national or private implementations or in the control of particular services. Each item of information is contained in an information element which has a name that describes its contents.

The first three Information Elements (or IEs) of every message are mandatory and are:

1. Protocol Discriminator – to identify the protocol
2. Call Reference – to identify the call to which the message refers
3. Message Type – to identify the message.

As these Information Elements (IEs) are mandatory for all messages, they have individual formats. The protocol discriminator always consists of one octet. The call reference consists of one or more octets, with the first octet indicating the length (in octets) of the call reference value which follows. The message type is formatted as a single octet except where the value in the first octet is hex '00'. This is the code for an escape to a national message set and the value for the national message is contained in the next octet. Figure 2-1 shows the layout of an ISDN Access Layer 3 message.

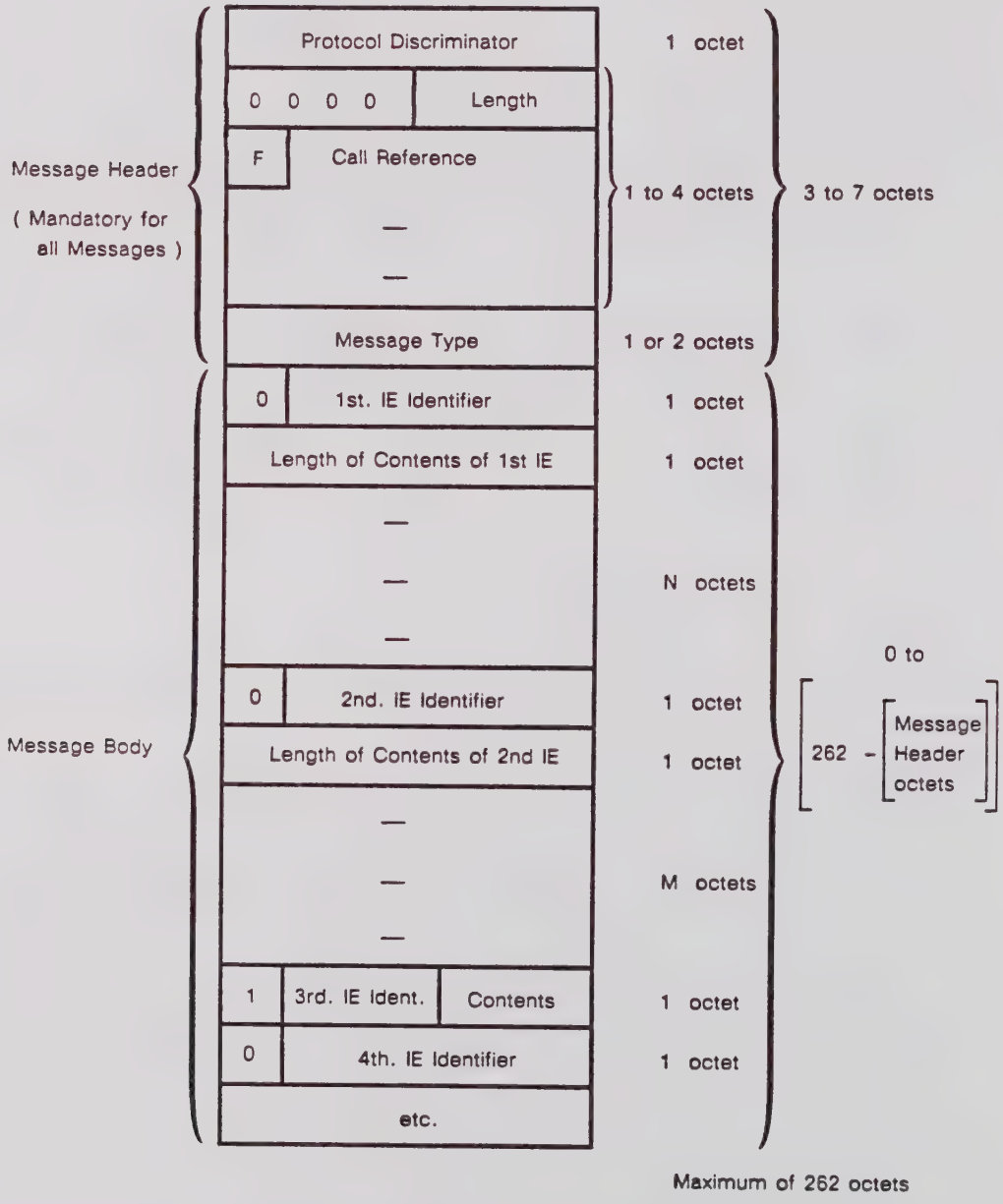


Figure 3-1: ISDN Access Layer 3 message format.
(Note: CCITT specifies a maximum of 260 octets per message.)

All other information elements have one of two defined formats. The simplest format is the single octet format. It is identified by a "1" in bit 8 of an information element identifier octet. This is followed by a 3 bit information element identifier field and a 4 bit contents field (see the 3-rd IE in Figure 3-1).

The other information element format is called the variable length IE format which consists of :

1. Information element identifier – of one octet
2. Length (N) of contents of information element – one octet
3. 'N' octets of information.

Every information element contains one or more fields in addition to the IE identifier and the length fields. Most fields are less than or equal to one octet in length, but several can be longer than one octet, for example, call reference, called number, etc. Every field can take on a range of values which have been assigned explicit, descriptive names as FORTH constants (see also ISDN TEST MANAGER Manual and the LAYER 3 (INTER)NATIONAL VARIANT Manual).

3.3. THE MENU BUILDING METHOD: AN OVERVIEW

The Menu Building Method is a menu driven message building, display, editing and storing utility which helps the user to create both correct and incorrect Layer 3 messages. In addition the messages created with this system can also be easily accessed using the programming interface to the Layer 3 Message Buffer Pool System as described in 6.7.

The user builds a message, information element by information element, after setting the octet and octet field values for each information element. A group of up to 60 Layer 3 messages can be created. This entire group of messages can be saved to disk and, at a later time, retrieved as required. Each message can be given a unique (user defined) name which can be used to address the message. Furthermore the use of a comment field associated with each message permits the user to briefly describe the contents of that message.

Messages need to be:

- built
- displayed
- edited and
- stored.

To be useful, the process of creating, examining, modifying and storing messages should be as simple as possible. In the Menu Building Method, each of these tasks has been allocated to a separate menu as shown in Figure 3-2. The solid lines connecting the menus in this figure represent the normal paths of access between menus in the Menu Building Method.

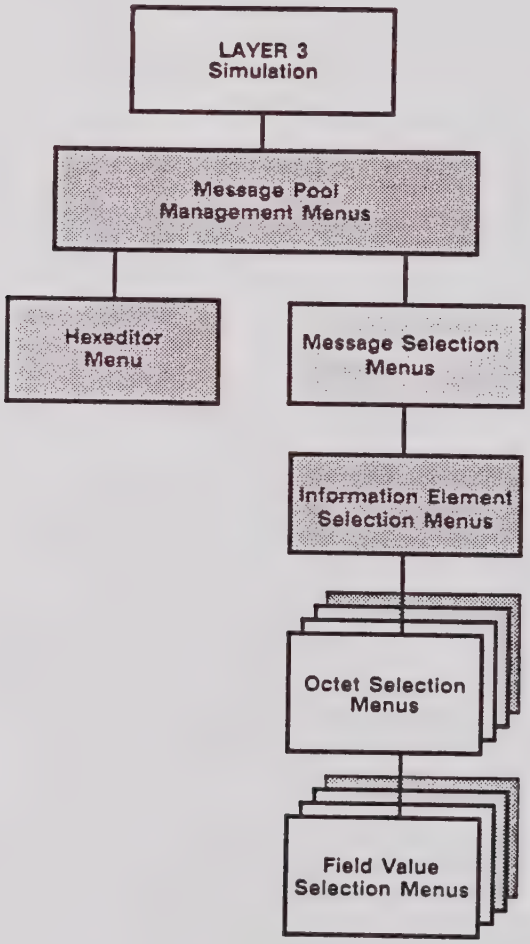
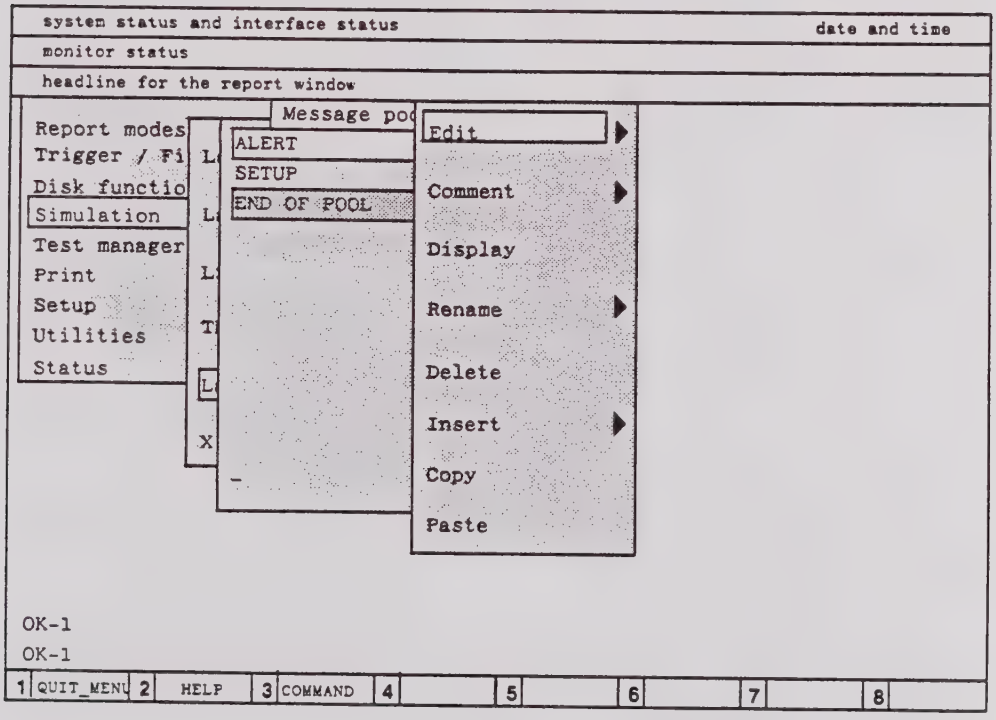


Figure 3-2: Message Building Menu Structure

The message building system is accessed by selection SIMULATION in the main menu and LAYER 3 MBS in the next menu. The menu appearing now shows the message pool contents. If the pool is empty, only the item END_OF_POOL (which is not a message) is visible. Every item in the pool menu is selectable. Stepping to the right of a selected menu item shows which actions can be applied to the selected item from the message pool.



MENU SELECTION:

Edit	: Edit the selected message either in the hexeditor or in the mnemonic editor
Comment	: Enter the comment field of the message
Display	: Display the message
Rename	: Rename the message
Delete	: Delete the message
Insert	: Insert a new (initially empty) message in front of the selected message
Copy	: Copy the selected message (will be put into a paste buffer)
Paste	: Insert the last copied message in front of the selected message

3.3.1. Message Building

The message building system is entered by selecting a message to edit from the message pool, selecting the EDIT function and then selecting the MNEMONIC option from the HEX/MNEMONIC menu. If the message you are editing already exists in the pool, the MBS will display the message fields directly and allow you to edit them.

There are two different modes to work with the message building system: a manual mode and an automatic mode. In the manual mode, the user selects a message and then selects information element by information element which shall be immediatly put into a Message Building Buffer. The automatic mode permits the user to select any required IEs (for a message type) without building each one into the Message Building Buffer individually. The IEs can be selected in any order but are sorted by the Message Building System into the correct order and neccessary SHIFT IEs are inserted when the message is built. This mode restricts the user to only one IE of each type per message type. In both modes, only one message at a time can be build. The user may choose the mode and parameters for the message header in the Message building system setup menu (section 3.4).

Message building begins in the Message Selection Menu (section 3.5) where the user chooses a message type and then enters the Information Element (IE) Selection Menu. This causes the Message Building Buffer to be cleared, and the message header (see Figure 3-1) to be built into the Message Building Buffer (in the manual mode). The Menu Building Method has default settings for all parameters that can be built into a message. Note that if these parameters are changed during message generation, then the new parameter values become the current default values. A detailed explanation of these features is given in section 3.4 .

In the Information Element (IE) Selection Menu (section 3.6) all mandatory IEs, for the selected message type, all optional IEs and further IEs which are not specified for the particular message type are listed.

The user can select an IE by positioning an arrowhead with the cursor control keys. If the IE contents are known to be those required, the user can build the IE into the Message Building Buffer immediately with the BUILD_IE function key. If the IE contents are unknown or require modification, the user can examine/change the component octets and field values of the selected IE via the Octet Selection Menu (section 3.7) before building the IE into the Message Building Buffer.

After all the required IEs have been built into the Message Building Buffer, the user can return to the Message Selection Menu where the message can be displayed or further processed using the Hexeditor or the Message Pool Management. The WRITE function key should be pressed to save the message into the selected item in the message pool and the QUIT_MBS function key should be pressed to leave the message building system without saving the generated message.

3.3.2. Message Display

Two distinct methods for display of the Layer 3 messages are available in the Menu Building Method.

It is possible to display the contents of a message using the hexeditor. A message is available in the hexeditor if it is saved in the message pool. The selected message is displayed in hex and the user has the option to modify the message (section 3.3.3).

A detailed display of the contents of a message is possible using the DISPLAY function in the Message Selection Menu or in the Message Pool Menu. The format of the message display may be selected in the Report Mode Selection menu (refer to ISDN MONITOR manual, section 4.5) or the corresponding commands.

3.3.3. Message Editing

The hexeditor allows the user to modify messages (either complete or partially complete) that have been stored and retrieved from disk to the Message Pool. The message contents are displayed (in hex) and can then be easily modified using the editing function keys (section 3.8).

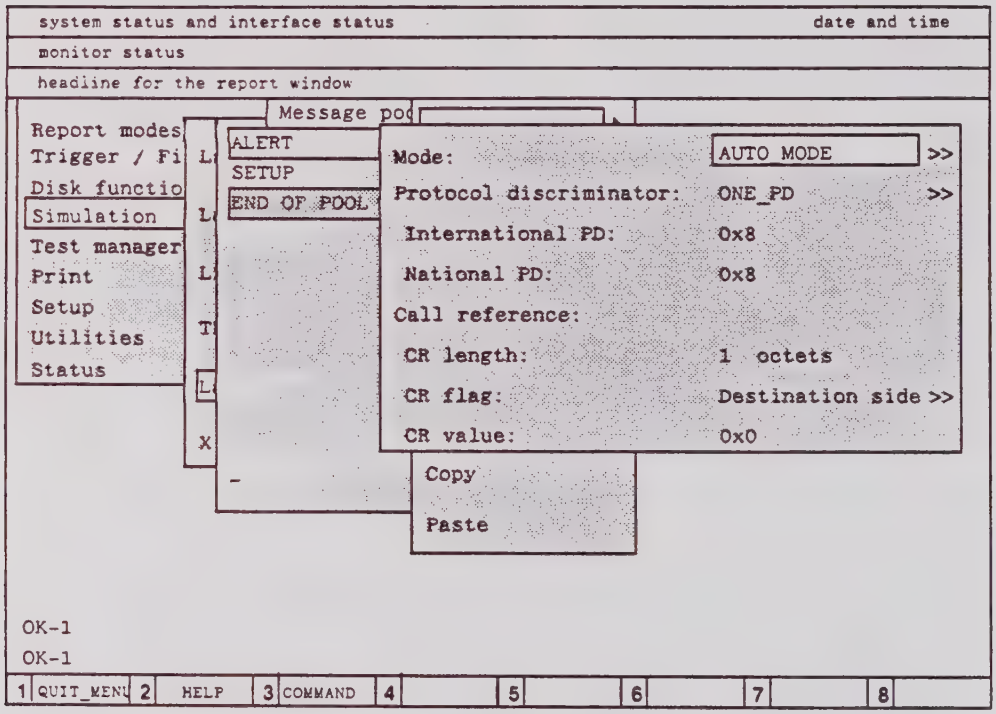
On completion of editing, the message can be (re)stored in the Message Pool.

3.3.4. Message Storage

The Message Pool Management Menu enables the user to store and retrieve between 30 and 60 Layer 3 messages as a single group (Message Pool) to/from disk. Only one Message Pool is available in the Message Building System at a given time although many Message Pools may be stored on disk. Individual stored messages may have an extension depending on their length. Each message has a name field and a comment field associated with it to enable easy identification of the messages. These fields can be changed by the user.

The Message Pool Management Menu displays the name fields (and the comment fields) of messages available in the Message Pool (see also section 3.3).

3.4. THE MESSAGE BUILDING SYSTEM SETUP MENU



MENU SELECTION:	
Mode	: Select manual/automatic mode of message generation
Protocol discriminator	: Select same or different PD for international and national msg
International PD	: Set international protocol discriminator
National PD	: Set national protocol discriminator
Call reference	:
CR length	: Set number of bytes used for call reference value
CR flag	: Set call reference flag
CR value	: Set call reference value

3.4.1. Changing the Call Reference Value

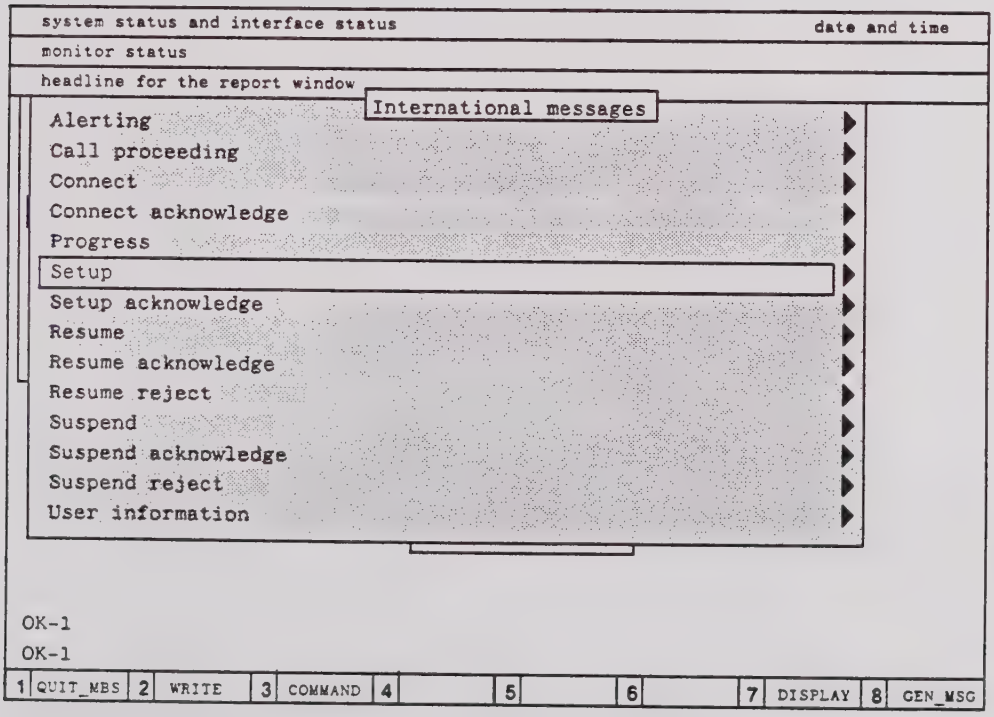
The call reference value can be set to any value between 0x00 and the maximum value allowed by the call reference length. The following table shows the maximum value allowed for each call reference length.

	Call Reference Length (bytes)			
	0	1	2	3
maximum Call Reference Value (hexadecimal)	no value accepted	0x7F	0x7FFF	0x7FFFFF

3.5. THE MESSAGE SELECTION MENU

The Message Selection Menu is accessed by selecting a message to edit mnemonic from the message pool and then selecting the International MSG or National MSG option. If the loaded layer 3 variant specifies more than one message set, different menus are available to access the messages.

This screen provides the means for a user to select a CCITT Layer 3 message type for construction. The selection is made by moving the arrowhead to the required message with the cursor control keys.



FUNCTION KEYS:

QUIT_MBS	: Quit message building system without saving the generated message in the pool
WRITE	: Save generated message into the selected item in the message pool and quit MBS
DISPLAY	: Display contents of message building buffer
GEN_MSG	: Generate the message (only available in auto mode)

3.5.1. Generating a Message in the manual Mode

3.5.1.1. Selecting a Message Type

The message type is selected by using the cursor up and down keys to point to the desired message (represented by the message mnemonic). Entering the Information Element Selection Menu (by pressing the cursor right key) starts the message building process by clearing the Message Building Buffer and building the protocol discriminator, call reference and message type into this buffer. The Information Element Selection Menu (see section 3.6) is then displayed.

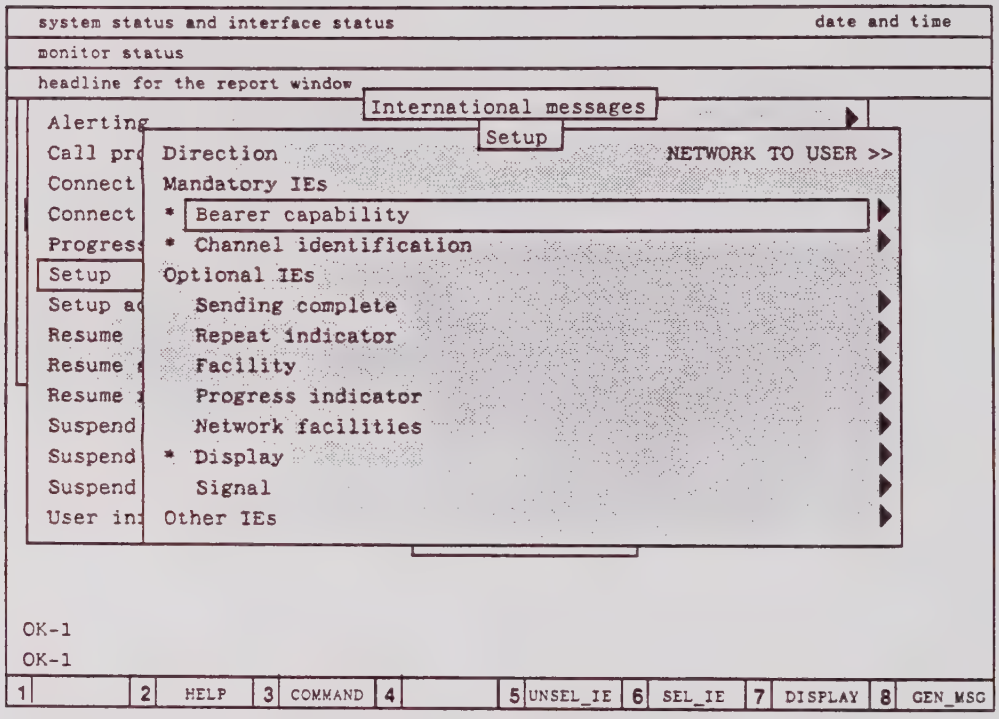
3.5.2. Generating a Message in the automatic Mode

After all IEs have been selected and/or modified (in the IE or other Selection Menus) the GEN_MSG function key is used to build both the message header and the message body into the Message Building Buffer. Thereafter the message can be treated in the same manner as messages generated in the manual mode.

3.6. THE INFORMATION ELEMENT SELECTION MENU

The Information Element (IE) Selection Menu permits the user to choose CCITT or other (national variant) IEs to be included in the Layer 3 message defined in the Message Selection Menu (section 3.5). IEs to be added to the Layer 3 message are selected by moving the arrowhead (with the cursor up and down keys) to the required IE.

For each message type, mandatory IEs, optional IEs and further IEs which are not specified by the recommendation for the selected message are listed. The ' * ' in front of the IE name indicates that the IE is selected for message generation (auto mode) or already build into the selected message (manual mode).



MENU SELECTION
AND FUNCTION KEYS:

Direction	: Select the message direction
OTHER IEs	: Enter menu of further Information Elements
UNSEL_IE	: Deselect the IE for building (automatic mode)
SEL_IE	: Select the IE for building (automatic mode)
UNDO_LAST	: Remove last IE built into the message building buffer (manual mode)
BUILD_IE	: Build IE using current octet contents (manual mode)
DISPLAY	: Display contents of message building buffer
GEN_MSG	: Generate the message (auto mode)

3.6.1. Building Information Elements (manual mode)

All IEs listed in the IE Selection Menu can be built into the Message Building Buffer. The cursor is positioned to point to the required IE and pressing the BUILD_IE function key causes this IE to be built into the Message Building Buffer. The IE will contain octets currently defined for inclusion in that IE. The current contents of the fields which are contained in the included octets will be used in the building process. There is no change of menu.

After an IE has been built into the Message Building Buffer, a ' * ' is printed in front of the IE name so that the user can keep track of which IEs have been built. Each IE can be built into the Message Building Buffer more than once if desired. The only limit to the number times an individual IE can be built is the capacity of the Message Building Buffer.

3.6.2. Selecting Information Elements (auto mode)

A particular IE is selected for message generation in auto mode by pressing the SEL_IE function key while the desired IE is selected by the cursor position. A ' * ' is printed in front of the IE name to indicated all selected IEs. To generate the message, the GEN_MSG key should be pressed.

3.6.3. Altering the Information Element Octets

If the user wishes to modify the contents of a particular IE, the cursor is first positioned to the required IE and the cursor right key is pressed. This will transfer the user to the Octet Selection Menu which lists all the octets and octet fields applicable to the selected IE. Thereafter individual octet fields can be selected and modified prior to returning to the IE Selection Menu.

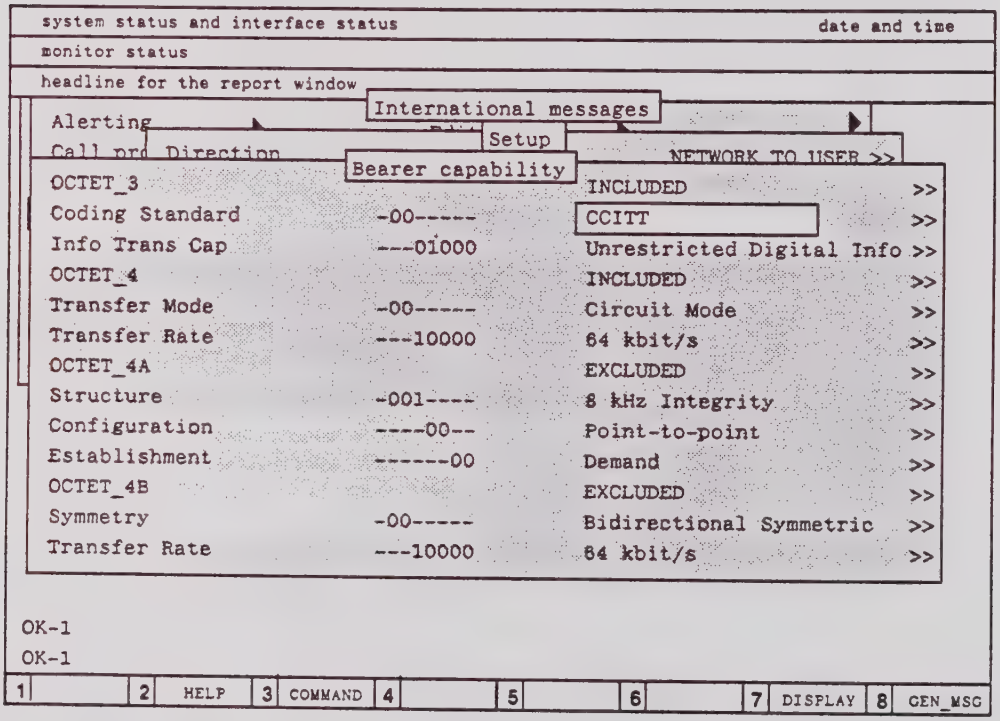
Note that in auto mode an IE is **automatically** selected for message generation by entering the Octet Selection Menu. If the IE should not be selected, the UNSEL_IE function key is to be pressed after returning to the IE Selection menu.

3.6.4. Return to the Message Selection Menu

The user can return to the Message Selection Menu by pressing the cursor left key. If the BUILD_IE function key was not used while in the IE Selection Menu the user is transferred to the Message Selection Menu without affecting the contents of the Message Building Buffer.

3.7. THE OCTET SELECTION MENU

The Octet Selection Menu permits the user to choose the octets for inclusion in an IE, and the octet fields to modify from a mnemonic list which is relevant to the selected information element. The sceen example below shows the list applicable to the "Bearer capability" information element within a "SETUP" message. Octets and octet fields are selected for display or modification by moving the cursor (with the cursor up and down keys) to the required octet or octet field.


system status and interface status
date and time
monitor status
headline for the report window
International messages
Alerting
Setup
Call prd Direction
NETWORK TO USER >>
Bearer capability
OCTET_3 INCLUDED >>
Coding Standard -00----- CCITT >>
Info Trans Cap ---01000 Unrestricted Digital Info >>
OCTET_4 INCLUDED >>
Transfer Mode -00----- Circuit Mode >>
Transfer Rate ---10000 64 kbit/s >>
OCTET_4A EXCLUDED >>
Structure -001---- 8 kHz Integrity >>
Configuration ----00-- Point-to-point >>
Establishment ------00 Demand >>
OCTET_4B EXCLUDED >>
Symmetry -00----- Bidirectional Symmetric >>
Transfer Rate ---10000 64 kbit/s >>
OK-1
OK-1
1 2 HELP 3 COMMAND 4 5 6 7 DISPLAY 8 GEN_MSG

FUNCTION KEYS:

DISPLAY	: Display contents of message building buffer
GEN_MSG	: Generate the message (auto mode)

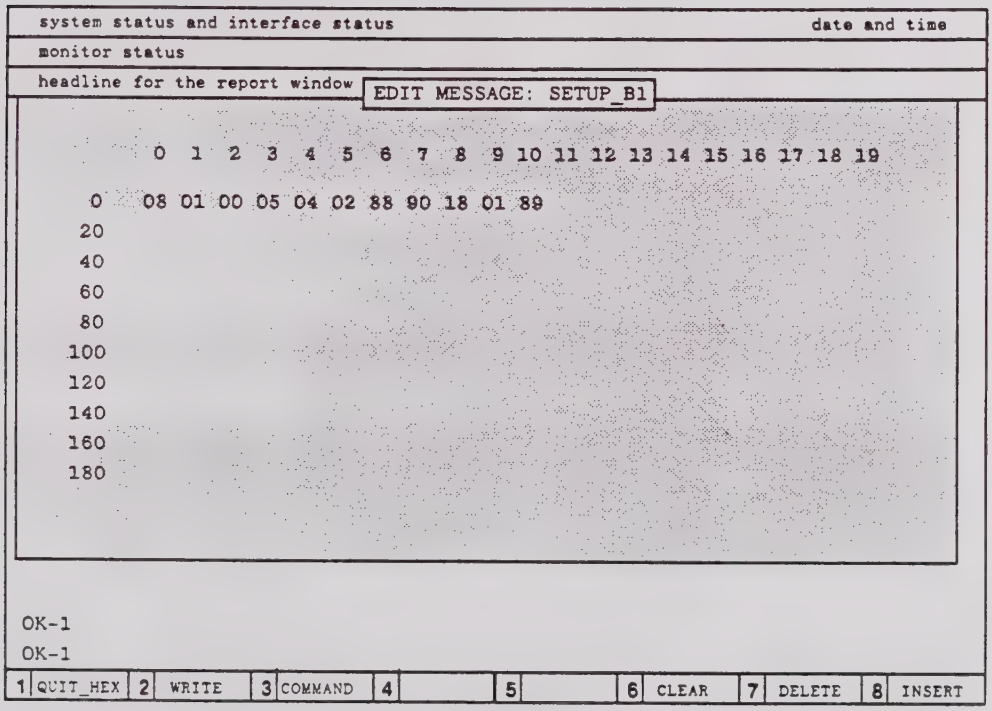
3.7.1. Modification of Octets and Fields into the Selected IE

Octets can be included in or excluded from the selected IE by menu selection. Furthermore the field contents of the octets may be modified.

Note that it is irrelevant whether an octet is first modified and then listed for inclusion, or listed for inclusion into the IE and then modified as it is not built into the Message Building Buffer until the BUILD_IE (manual mode) or GEN_MSG (automatic mode) function key is used.

3.8. THE HEXEDITOR MENU

The hexeditor allows the user to edit messages in hexadecimal format. The hexeditor is entered by selecting a message to edit from the message pool, selecting the EDIT function and then selecting the HEX EDIT option from the HEX/MNEMONIC menu. The contents of the hexeditor may be altered by inserting, replacing or removing information.



3.8.1. Editing the contents of the Editor Buffer

The contents of the Editor Buffer can be edited by moving the cursor to the byte where the change is required and performing one of the following actions:

Typing: If a valid HEX character is entered then this appears as the most significant nibble of the byte that the cursor is on. A small flashing cursor appears indicating that input for the least significant nibble is required. The byte may not be left until this second HEX character is given. If a valid character is entered then the small flashing cursor is turned off and the cursor moves on to the next byte.

Delete Byte: The byte at the cursor position can be removed by pressing the DELETE function key.

Insert Byte: By pressing the INSERT function key, a byte of zero is inserted at the cursor position.

Clear Buffer: The hexeditor buffer is cleared by pressing the CLEAR function key.

3.8.2. Leaving the hexeditor

The hexeditor menu may only be left by pressing the function key QUIT_HEX or WRITE which have the following meaning:

QUIT_HEX — terminates the hexeditor returning to the HEX/MNEMONIC menu. The hexeditor contents will not be saved in the message pool.

WRITE — terminates the hexeditor returning to the HEX/MNEMONIC menu. The selected item of the message pool will be updated by the hexeditor contents.

3.9. THE MESSAGE POOL MANAGEMENT

The message pool management menu and the associated functions are already describes in section 3.3. The following function keys are available from the message pool menu:

1	QUIT_MENU	2	HELP	3	COMMAND	4		5	SAVE POOL	6	LOAD POOL	7	CLR POOL	8	CONCAT

SAVE POOL — A menu is entered to save the contents of the message pool on a disk file.

LOAD POOL — A menu is entered to load a message pool from disk.

CLR POOL — All entries of the message pool will be cleared.

CONCAT — A menu is entered to concatenate a message pool from disk to the existent message pool.

4. LAYER 2 EMULATION

4.1. SEND COMMANDS

4.1.1. COMMANDS EXECUTED BY INVOKING THE STATE MACHINE

The commands in this section are executed under the control of the L2 State Machine. All frames sent as a result of these commands use values for NR, NS, P and F that are determined totally by the State Machine. The current values of these variables given by the user prior to executing the commands are ignored, even though they may have been changed by the commands in Section 4.3.1. However, the SAPI & TEI values that are sent will be the current values.

4.1.1.1. LINK LAYER STATE MACHINE

SABME **(--)**
A SABME frame will be generated and the emulation will expect a UA response. When the UA is received, the emulation will be in the in the Multiple frame established state (L2 state 7.0). The emulation must be in the Multiple frame established state before any packets can be sent or received.

DISC **(--)**
A DISC frame will be sent. The emulation will be expecting a UA response. When the UA is received, the emulation will go into the Identity assigned state (L2 state 4).

DM **(--)**
A DM frame will be generated. The emulation will be forced into the disconnected state.

UI_M / UI_3 **(address --) or (--), see section 4.3.2**
A UI frame will be generated (M: Management or ASP layer originated frame; 3: Layer 3 originated frame). The UI frame is transparent for the state machine. No flow control or rejection mechanisms are active.

SENDP **(address --) or (--), see section 4.3.2**
Send a user defined data packet. The data bytes sent will be transmitted in an I frame with the correct sequence values.

4.1.1.2. MANAGEMENT LAYER STATE MACHINE

IDENTITY_REQUEST **(TEI --)**
Sends the TEI management frame 'Identity request' with the given TEI as the Action Indicator Ai. Reference number Ri is a random number generated by the state machine. After sending the frame the State machine expects an answer 'Identity assigned' or 'Identity denied') from the network side. The possible retransmission is made according to CCITT TEI assignment procedure. A new CES is chosen for the procedure. If there are no CES's available, no action is taken. Can be used only in the user emulation.

IDENTITY_ASSIGNED **(Ri \ TEI --)**
The TEI management frame 'Identity assigned' is sent with the given values of Ri and Ai as if an 'Identity request' were received. A new CES is reserved for the connection and the layer 2 state machine is set to state 4 (Identity assigned). Can be used only by the network side.

IDENTITY_DENIED **(Ri \ TEI --)**
'Identity denied' message is sent to the user side as if an 'Identity request' with an Ai value to be denied were received.

IDENTITY_CHECK_REQ **(TEI --)**
Causes 'Identity check request' to be sent and Identity check procedure to be iniatiated. All necessary procedures according to CCITT are made including possible Identity removals. Only for the network side.

IDENTITY_CHECK_RESP **(TEI --)**
The user side sends an 'Identity check response' message with the given TEI as an Ai as if an 'Identity check request' were received.

IDENTITY_REMOVE **(TEI --)**
The network side sends an 'Identity remove' message twice and removes the TEI value(s) from the link and frees the associated CES for later use.

IDENTITY_VERIFY **(TEI --)**
The TEI management message 'Identity verify' is sent by the user side and the Identity verify procedure is started. If there is no response from the network side after retransmission, the TEI value will be removed.

XIDC **(--)**
Initiates the XID procedure. Sends out an XID command frame (P=0), which has the XID parameters (for N201TX (transmit), N201 (receive), K, T200) set to the K1197 preferred values. It starts the XID Timer (TM20), clears the XID retransmission counter (RC20), and goes to ML State 3, if the User side (or stays in ASP State 1, if the Network side).

4.1.2. COMMANDS EXECUTED OUTSIDE THE CONTROL OF THE STATE MACHINE

The commands in this section send frames outside of the control of the State Machine. These frames use the existing (stored or given by the user) values of SAPI, TEI, NR, NS, P, F, W, X, Y, and Z, as needed. Because these values can be manually changed by the commands in Section 4.3.1, any, or all of the values might differ from what they would be if the frames were sent under control of the State Machine. Since these frames might therefore conflict with what the peer Layer 2 is expecting (i.e. P, F, NS, NR, SAPI, TEI etc.), unexpected results could occur and caution is advised.

S_SABME **(--)**
Sends a SABME command

S_DISC **(--)**
Sends a DISC command

S_UA **(--)**
Sends a UA response

S_FRMR **(--)**
Sends a FRMR response

S_DM **(--)**
Sends a DM response

S_RR **(--)**
Sends an RR response

S_RNR **(--)**
Sends an RNR response

S_REJ **(--)**
Sends an REJ response

S_RRC **(--)**
Sends an RR command

S_RNRC **(--)**
Sends an RNR command

S_REJC **(--)**
Sends an REJ command

S_I **(address --) or (--)**
Sends an I frame, see section 4.3.2

S_UI **(address --) or (--)**
Sends a UI frame, see section 4.3.2

SENDF **(address --) or (--), see section 4.3.2**
Sends a user defined string as a Layer 2 frame. This frame will not have the proper header bytes (address and control) automatically, as it is the case for the other commands in this section. The header in this case is constructed directly from the first 3 bytes of the given data field.

SEND_FRMR	**(MDL error code --)** Sends an FRMR frame with proper W, X, Y and Z values. The other parameters are the current values of the state variables, last received frame data etc. The use of this frame is always done in conjuction with an MDL ERROR CODE defined by CCITT Q.921 APPENDIX II TABLE II-1. The error codes J, L, N and O causes an FRMR to be sent while the other codes are ignored. The format and meaning of the codes are explained in the chapter 4.3 SETTING VARIABLES (ERROR CODES). (See also 6.5.)
S_ID_REQUEST	**(TEI --)** Sends TEI management frame 'Identity request' with the given TEI value as Action Indicator Ai.
S_ID_ASSIGNED	**(Ri \ TEI --)** Sends TEI management frame 'Identity assigned' with the given values as Action Indicator Ai and Reference number Ri.
S_ID_DENIED	**(Ri \ TEI --)** Sends TEI management frame 'Identity denied' with the given value as Action Indicator Ai and Reference number Ri.
S_ID_CHECK_REQ	**(TEI --)** Sends TEI management frame 'Identity check request' with the given TEI value as Action Indicator Ai.
S_ID_CHECK_RESP	**(TEI --)** Sends TEI management frame 'Identity check response' with the given TEI value as Action Indicator Ai.
S_ID_REMOVE	**(TEI --)** Sends TEI management frame 'Identity remove' with the given TEI value as Action Indicator Ai.
S_ID_VERIFY	**(TEI --)** Sends TEI management frame 'Identity verify' with the given TEI value as Action Indicator Ai.
S_XID	**(--)** Sends an XID response
S_XID0	**(--)** Sends an XID response with zero length I field
S_XIDC	**(--** Sends an XID command
S_XIDC0	**(--)** Sends an XID command with zero length I field

Note: S_XID and S_XIDC cause the current contents of the 4 XID frame variables (for N201, N201TX, K, and T200) to be sent in the XID frame.

4.2. SERVICE PRIMITIVES

The commands in this section are executed under the control of the L2 State Machine. The comments in Section 4.1.1 also apply to these commands.

DL_ESTABLISH **(--)**
Establishes a Layer 2 connection. If a TEI was not assigned, the TEI assignment procedure is executed (User emulation only). Then an SABME is sent and the emulation waits for a UA response. (Same action as SABME, Section 4.1.1.)

DL_RELEASE **(--)**
Releases a previously established Layer 2 connection.
(Same action as DISC, Section 4.1.1).

DL_SET_BUSY **(--)**
If the Layer 2 State Machine is not already in one of the 'OWN RECEIVER BUSY' states, it sends out an RNR (F=0) frame. The Layer 2 then goes into one of the 'OWN RECEIVER BUSY' states. (If the Layer 2 is already in one of the 'OWN RECEIVER BUSY'states, no action will occur). If the peer Layer 2 sends a P=1 command, the own Layer 2 responds with an RNR response.

DL_CLR_BUSY **(--)**
If the Layer 2 State Machine is in one of the 'OWN RECEIVER BUSY' states, it sends out an RR (F=0) frame. The Layer 2 then exits from the 'OWN RECEIVER BUSY' state to a state where it can accept I frames. If the peer Layer 2 sends a P=1 command, the local Layer 2 responds with an RR response.

DL_DATA **(address --) or (--), see Section 4.3.2**
Sends a Layer 3 packet using acknowledged operation.
(Same action as SENDP, Section 4.1.1).

DL_UNIT_DATA **(address --) or (--), see Section 4.3.2**
Sends a Layer 3 packet using unacknowledged operation.
(Same action as UI_3, Section 4.1.1).

DL_DATA_REQ **(addr / length --)**
Same as DL_DATA, but transfers frames longer than 255 Bytes.

DL_UNIT_DATA_REQ **(addr / length --)**
Same as DL_UNIT_DATA, but transfers frames longer than 255 Bytes.

MDL_UNIT_DATA **(address --) or (--), see Section 4.3.2**
Send a ML (or ASP) packet using unacknowledged operation.
(Same action as UI_M, Section 4.1.1).

MDL_ASSIGN_R	**(TEI --)** The management entity requests that the data link layer associates the TEI value on the stack with the currently active connection endpoint. (To be used ONLY in states 1 or 3 for the Network, 2 or 3 for the User).
MDL_ASSIGN_I	**(--)** The data link layer indicates to the management entity the need for any TEI value.
MDL_REMOVE	**(--)** The management entity requests the current data link layer to remove the association between its TEI value and its Connection Endpoint Suffix.
MDL_EXISTED	**(SAPI -- CES / 1) or (SAPI -- 0)** If any data link is associated with the given SAPI value and is active (e.g. the L.2 state is not idle), the Connection Endpoint Suffix of that data link is returned together with a success flag.
MDL_ALLOCATE	**(SAPI -- CES / 1) or (SAPI -- 0)** A free data link controller is searched and if successful, its SAPI is set to the given value and the Connection Enpoint Suffix is returned togethter with a success flag.
MDL_DEALLOCATE	**(CES --)** The SAPI and TEI values of the data link layer with the given Connection Endpoint Suffix are released.
MDL_SELECT	**(CES / SAPI --)** A data link associated with the given Connection Endpoint Identifier is searched, if found this link is set to be the currently active link.
MDL_CURRENT	**(-- CES / SAPI)** The Connection Endpoint Identifier of the currently active data link is returned.
PH_ACTIVATE	**(--)** (For Basic Access only) The S-Bus is activated.
PH_DEACTIVATE	**(--)** (For Basic Access only) The S-Bus is deactivated.

4.3. SETTING VARIABLES

4.3.1. DATA LINK LAYER CONTROLLING VARIABLES

The commands in this section enable the user to change values of the Layer 2 State Machine parameters. Note that frames that are sent from the State Machine ignore the current values of NR, NS, P, F, W, X, Y, and Z because any required values are generated totally by the State Machine. (e.g. by the commands defined in Sections 4.1.1 and 4.2) Only the commands defined in Section 4.1.2 will use the current parameter values.

Note: It is important to carefully observe the fundamental requirement of the FORTH language: A number **MUST** be on the stack before executing any command that expects a numerical input, as indicated in some of the following commands by **(n --)**.

Note: In the following list of words to change various variables there is often two words for the same kind of action – one that begins with PT-... and the other one without this beginning (eg. T200 and PT-T200). The version **with PT-...** changes the 'default' value, that is, the value of the **constant** which is used always at the basic initialization of the system to give the value for the variable(s). An initialization could be rebooting the whole K1197 system.

The word **without PT-...** changes only the value of the **variable**. This variable is also unique for each CES, that is, every CES can have its own value for the variable (for example CES=0 can have T200 = 1 sec, CES=1 T200 = 1,5 sec etc.). These values are set to the default values (ie. the value of the **current** constant value) always when initializing an operating mode.

STATE VARIABLES

VS	(n --)	Sets V(S)
VA	(n --)	Sets V(A)
VR	(n --)	Sets V(R)

SEQUENCE NUMBERS

NR	(n --)	Sets N(R)
NS	(n --)	Sets N(S)

COUNTERS

RC	(n --)	Sets RC
RC20	(n --)	Sets the value of the XID retransmission counter
RC201	(n --)	Sets RC201

WINDOW

K	(n --)	Sets the window size K
PT-K	(n --)	Sets the default value for the K1197 for the window size K

STATES

L2ST	(n --)	Sets the state of the data link layer (the link that is selected is determined by the Connection Endpoint Suffix CES).
MLST	(n --)	Sets the state of the USER Management Layer.
ASPST	(n --)	Sets the state of the NETWORK Assignment Source Point.

Note: see APPENDIX A for the definitions of the state numbers.

FLAGS

U/N	(n --)	Selects USER emulation if n = **TE**; NETWORK if n = **NT**.

TIMERS

TM20	(n --)	Sets the duration of the XID procedure timer TM20 (n is in tenths of a second); default value = 2,5 sec.
T200	(n --)	Sets the duration of the primary timer T200 (n in tenths of a second); automatically sets T201=T200 and T202=2*T200; default value = 1 second.
PT-T200	(n --)	Sets the preferred duration, for the K1197, of the primary timer T200 (n in tenths of a second); automatically sets T201=T200 and T202=2*T200; default value = 1 second.
T201	(n --)	Sets the duration of timer T201 (minimum time between network retransmission of the TEI-identity check messages); default value = 1 second.
T202	(n --)	Sets the duration of timer T202 (minimum time between user transmission of the TEI-identity request messages); default value = 2 seconds.
T203	(n --)	Sets the duration of timer T203 (maximum time allowed without frames being exchanged); default value = 10 sec.
PT-T203	(n --)	Sets the preferred duration, for the K1197, of timer T203 (maximum time allowed without frames being exchanged); default value = 10 sec.

MAXIMUM NUMBERS

NM20	(n --)	Sets the maximum number of XID command frame retransmissions; default value = 3.
N200	(n --)	Sets the maximum number of retransmissions of a frame due to expiry of T200; default value = 3.
N201	(n --)	Sets the maximum number of octets allowed in a **received** UI or I frame information field; default value = 260. (maximum value permitted = 286)
PT-N201	(n --)	Sets the default value for the K1197 for the maximum number of octets allowed in a **received** UI or I frame information field; default value = 260 (maximum value permitted = 286).
N201TX	(n --)	Sets the maximum number of octets allowed in a **transmitted** UI or I frame information field; default value = 260. (maximum value permitted = 286)
PT-N201TX	(n --)	Sets the default value for the K1197 for the maximum number of octets allowed in a **transmitted** UI or I frame information field; default value = 260. (maximum value permitted = 286)
N202	(n --)	Sets the maximum number of consecutive TEI assignment requests, due to expiry of T202 (ie: occurs when the Network does not respond to an ID request); default value = 3.

P & F BITS

P0	(--)	Reset poll bit
P1	(--)	Set poll bit

Note: The P bit is overwritten by any command transmitted.

F0	(--)	Reset response bit
F1	(--)	Set response bit

Note: The F bit is overwritten by any received frame.

FRMR BITS

W0	(--)	Reset the W bit
W1	(--)	Set the W bit
X0	(--)	Reset the X bit
X1	(--)	Set the X bit
Y0	(--)	Reset the Y bit
Y1	(--)	Set the Y bit
Z0	(--)	Reset the Z bit
Z1	(--)	Set the Z bit

I/UI FRAMES

PKT	(n --)	Stores a number (1 to 8) in the S-DATA-AUTO* variable. (See 4.3.2 for instructions how to operate with PKT and MAKE_DATA..)
MAKE_DATA1 / MAKE_DATA2 / MAKE_DATA3 / MAKE_DATA4 / MAKE_DATA5 / MAKE_DATA6 / MAKE_DATA7 / MAKE_DATA8	**(string address --)**	Permits the creation of up to 8 different data fields using either ASCII or hex characters. (See Section 4.3.2)
BYTES	(n --)	Sets the number of bytes in an I frame when they are sent using "SEND I-frame" in layer 2 simulation menu (section 2.1.3).
PKTS	(n --)	Sets the number of packets (I frames) to be sent when they are sent using "SEND I-frame" in layer 2 simulation menu (section 2.1.3).

MDL-ERROR-CODES defined by CCITT:

	ERROR CODE	ERROR CONDITION
Receipt of unsolicited response:	A	Supervisory (F=1)
	B	DM (F=1)
	C	UA (F=1)
	D	UA (F=0)
	E	Receipt of DM response (F=0)
Peer initiated re-establishment:	F	SABME
Unsuccessful retransmission (N200 times):	G	SABME
	H	DISC
Other:	I	Status Enquiry
	J	N(R) Error
	K	Receipt of FRFM response
	L	Receipt of non-implemented frame
	M	Receipt of I field not permitted (never used)
	N	Receipt of frame with wrong size
	O	N201 Error

SENDING FRAMES WITH ERROR

CRC_ERROR	(--)	After this command is issued, the next frame will be sent with an invalid CRC. Following frames will be sent with the correct CRC.
DO_ABORT	(--)	After this command is issued, the next frame will be sent and aborted during transmission. Following frames will be sent correctly.
GOOD_CRC	(--)	After this commands is issued, the next frame will be sent with the correct CRC. This is the default setting for transmission of all frames unless the CRC_ERROR or DO_ABORT command has been entered.

MISCELLANEOUS

SAPI	(n --)	Sets current SERVICE ACCESS POINT IDENTIFIER
TEI	(n --)	Sets current TERMINAL ENDPOINT IDENTIFIER.
CES	(n --)	Sets the current CONNECTION ENDPOINT SUFFIX (ensure n = 0..7). A CES is a data link, ie. it can be understood as an individual TE (at the user side).
CES=0	(--)	Sets the Connection Endpoint Suffix 0
CES=1 .. 7	(--)	Sets the Connection Endpoint Suffix 1..7 respectively.
ATEI	(n --)	Assign TEI. Sets the TERMINAL ENDPOINT IDENTIFIER and changes the Layer 2 and ML to the TEI assigned states.
XIDV	(N201TX / N201 / K / T200 --)	Stores the 4 values into the XID variables. These are the parameter values of the next XID frame that is sent outside of the Layer 2 State Machine. They are overwritten by any received (and valid) XID frame and by any XID frame sent via the State Machine.
FER	(m --)	Sets the Frame Error Rate that the receiver will simulate on received frames; also sets the BER=0. (example: FER = 10 => every 10th frame contains an error; it will be ignored.)
BER	(M / P --)	Sets the Bit Error Rate that the receiver will simulate on received frames. BER=Mx10E-P; also sets the FER=0. (example: BER = 5 x 10 EXP -3 => every 200th bit is erroneous; the frame will be ignored.) (M must be 0...9 and P 3...9)
CLREC	(--)	Clear Error Codes. Clears the table where the amount of created MDL ERROR CODEs are stored.
NEW_TEI	(--)	Sets at the user side the TEI value of the current CES to 127, which forces the CES to initialize Identity assignment procedure when it for the next time establishes a data link connection.

4.3.2. DATA BUFFER DEFINITIONS

The information packets that are sent as a result of executing the **Packet Commands** (i.e. UI_M, UI_3, SENDP, S_I, S_UI, SENDF, DL_DATA, DL_UNIT_DATA, or MDL_UNIT_DATA) must first be created by using the following commands:

MAKE_DATA1 **(string address --)**
Makes (defines) the data field #1. Prior to executing this command, the data field must be entered via the keyboard (i.e. " xxxxx" or X" xxxxxx" for hexadecimal characters), which places the address of the string on the stack. Used when the variable S-DATA-AUTO* is set to 1 (using PKT) and a Packet Command is executed.

MAKE_DATA2 / MAKE_DATA3 /

MAKE_DATA4 / MAKE_DATA5 /

MAKE_DATA6 / MAKE_DATA7 /

MAKE_DATA8 **(string address --)**
Defines data field #2, #3, #4 , #5, #6, #7, #8, respectively.

S-DATA-AUTO* Variable used to determine which of the above data fields are to be sent. Its value is set by using the command **PKT** . (If = 0, the direct keyboard data field is selected).

S-DATA-AUTO* = 0:	Keyboard input data field is sent
S-DATA-AUTO* = 1:	Data field #1 is sent
S-DATA-AUTO* = 2:	Data field #2 is sent
S-DATA-AUTO* = 3:	Data field #3 is sent
S-DATA-AUTO* = 4:	Data field #4 is sent
S-DATA-AUTO* = 5:	Data field #5 is sent
S-DATA-AUTO* = 6:	Data field #6 is sent
S-DATA-AUTO* = 7:	Data field #7 is sent
S-DATA-AUTO* = 8:	Data field #8 is sent

If the variable S-DATA-AUTO* has been set to 0, the user **MUST** precede the Packet Commands with a string from the keyboard (i.e.: " xxxxx"). Examples for creating and sending packets:

a. To create Data Field #3:
TYPE: " 1234ABCD" MAKE_DATA3 <CR>

b. To create Data Field #2 using hex characters:
TYPE: X" 02810A01999A9B9C" MAKE_DATA2 <CR>

c. To cause Data Field #2 to be sent:
TYPE: 2 PKT <CR>
UI_M <CR> (or any other Packet Command)

d. To send a Data Field directly from the keyboard:
TYPE: 0 PKT <CR>
" 1234ABCD" UI_M <CR> (or any other Packet Command)

4.4. MISCELLANEOUS

The commands in this section enable the user to change the mode of Emulation and to control operation of the L2, ML or ASP state machines. To allow total user control, the state machines can be turned on and off. If a particular state machine is turned off, it is possible to run this state machine step by step (e.g. for a particular event). See also the ISDN TEST MANAGER Manual.

4.4.1. EMULATION MODES

USER_EMUL	(--)	Enables the K1197 for the User emulation mode and initializes all Layer 2 variables.
NTWK_EMUL	(--)	Enables the K1197 for the Network emulation mode and initializes all Layer 2 variables.
PTP	(--)	Point-to-point: Enables the K1197 for the point-to-point configuration and initializes all relevant Layer 2 variables (only available for basic access).
PTMP	(--)	Point-to-multipoint: Enables the K1197 for the point-to-multipoint configuration and initializes all relevant Layer 2 variables (only available for basic access). This configuration is default.
?PTP	(-- flag)	Returns a true flag if the point-to-point configuration is selected, a false flag otherwise (always true for primary rate access and V/X interface).

4.4.2. EMULATION CONTROL

The following commands have been added to make it possible for the user to turn on and off the L2 and ML/ASP state machines. When the state machine is off, execution of the next event in the event queue is possible by using L2_EXEC / ML_EXEC.

The commands in this section are only considered to be useful in a Test Manager environment.

L2_ON	(--)	Turns the L2 state machine on. When turned on the automatic emulation handles all layer 2 events.
L2_OFF	(--)	Turns the L2 state machine off. When turned off the user has to take care of the layer 2 events, for example by using the commands specified in 4.1.2. See the note !
L2_EXEC	(--)	Executes the L2 state machine for the event in the layer 2 event queue. Can be used when L2 State machine is switched off to handle the latest (only one) L2 event according to the rules of the State machine.
?L2_ON	(-- flag)	Returns a true flag if the L2 state machine for the current Connection Endpoint Suffix (CES) is on, a false flag otherwise.
ML_ON	(--)	Turns the ML (User side) or ASP (Network side) state machine on.

ML_OFF	(--)	Turns the ML (User side) or ASP (Network side) state machine off. See the note!
ML_EXEC	(--)	Executes the ML (USER side) or ASP (NETWORK side) state machine for the event in the ML/ASP event queue.
?ML_ON	(-- flag)	Returns a true flag if the ML/ASP state machine for the current Connection Endpoint Suffix (CES) is on, a false flag otherwise. (There is only one ASP state machine on NETWORK side)

Note: The L.2 and ML/ASP state machines affect each other. For example, if L.2 is off a TEI request is not routed to the ML/ASP state machine. Therefore, for a correct behaviour of ML/ASP procedures the L.2 state machine must be "ON".

4.4.3. TRACING OF EMULATION EVENTS

L2_STATE_TRC

(trace flag --)
When switched on every layer 2 state transition is displayed as a trace report separately for each CES.

1 L2_STATE_TRC --> Trace on
0 L2_STATE_TRC --> Trace off (default)

L2_TIMER_TRC

(trace flag --)
When switched on every layer 2 timer action (start, stop, expiry) is displayed as a trace report separately for each CES.

1 L2_TIMER_TRC --> Trace on
0 L2_TIMER_TRC --> Trace off (default)

MDL_ERROR_TRC

(trace flag --)
When switched on, on every MDL-ERROR-INDICATION occurrance the corresponding MDL ERROR CODE is displayed as a trace report separately for each CES.

1 MDL_ERROR_TRC --> Trace on
0 MDL_ERROR_TRC --> Trace off (default)

4.4.4. COMMANDS FOR PRIMARY RATE ACCESS

The following commands are only available for K1197 with primary rate interface. They allow to set particular bits and alarms on the layer 1.

INT_BIT=1 (--)
Set the transmitted international bit in timeslot 0
(in PCM-30 framing format)

INT_BIT=0 (--)
Clear the transmitted international bit in timeslot 0
(in PCM-30 framing format)

SI1_BIT=1 (--)
Set the transmitted SI1 Bit in timeslot 0
(bit 1 in frame 13 of the multiframe in CRC-4 framing format)

SI1_BIT=0 (--)
Clear the transmitted SI1 bit in timeslot 0
(bit 1 in frame 13 of the multiframe in CRC-4 framing format)

SI2_BIT=1 (--)
Set the transmitted SI2 Bit in timeslot 0
(bit 1 in frame 15 of the multiframe in CRC-4 framing format)

SI2_BIT=0 (--)
Clear the transmitted SI2 bit in timeslot 0
(bit 1 in frame 15 of the multiframe in CRC-4 framing format)

=NAT_BITS **(national bits --)**
Set the transmitted national bits in timeslot 0
(in CRC-4 or PCM-30 framing format)

=ICHAR **(idle character --)**
Set the transmitted idle character

FERR=ODD (--)
Set the CEPT CRC-4 framing bits at odd bit location

FERR=EVEN (--)
Set the CEPT CRC-4 framing bits at even bit location

BALM_ON (--)
Turn on generation of blue alarm

BALM_OFF (--)
Turn off generation of blue alarm

YALM_ON (--)
Turn on generation of yellow alarm

YALM_OFF (--)
Turn off generation of yellow alarm

4.4.5. COMMANDS FOR BASIC ACCESS

The following commands are only available for K1197 with basic access interface. They allow to configure the interface.

S_BUS_SHORT (--)
Set short S-bus configuration

S_BUS_EXT (--)
Set extended S-bus configuration

S_BUS_PTP (--)
Set point-to-point S-bus configuration

IMPED=100 (--)
Set impedance to 100 ohm, i.e. turn on termination resistors on the connector module

IMPED=HIGH (--)
Set high impedance, i.e. turn off termination resistors on the connector module

POWER_FEED_ON (--)
Turn on power feeding

POWER_FEED_OFF (--)
Turn off power feeding

POWER_FEED_INVERS (--)
Set invers power feeding

EXTERN_CLK (--)
Set clocking mode to external clocking, i.e. the interface gets its clocking from an external clock source.
Attention: Be sure that an external clock source is applied **before** this command is executed ! Otherwise the application will lose its clock.

INTERN_CLK (--)
Set clocking mode to internal clocking, i.e. the application gets its clock from the own (internal) clock source of the interface. (This mode is default.)

SLAVE_CLK (--)
Set slave clocking mode, i.e. the application gets its clocking from the other interface.
Attention: Be sure that another interface is installed which is not in the slave clocking mode.

4.4.6. COMMANDS FOR V/X INTERFACE

The following commands are only available for K1197 with V/X interface. They allow to configure the interface.

NEW_SPEED	**(speed value--)** Set interface speed in bits per second.
EXTERN_CLK	**(---)** Select external clock source.
STANDARD_CLK	**(---)** Select the modem provided clocks as transmit and/or receive clocks.

4.5. LIST OF COMMUNICATION VARIABLES

The ISDN Test Pak utilizes variables to hold the decoded information from received frames and also for building headers, etc. for transmitted frames. There is a list of these variables below. They are available for the user to examine, manipulate, alter etc.

Note: Many of these variables can be altered by using the commands listed in Section 4.3.1. Checking/Changing the variables is only considered to make sense in a Test Manager environment, care must be taken when changing the values manually.

4.5.1. TRANSMIT VARIABLES

USR*NET	(-- addr)	TE emulating terminal NT emulating the network
STATE-L2*	(-- addr)	Layer 2 state variable (contains 0 if state machine is off). (See APPENDIX A)
VTEI*	(-- addr)	TEI assigned to the link
SAPI*	(-- addr)	SAPI assigned to the link
VS*	(-- addr)	V(S)
VR*	(-- addr)	V(R)
P*SENT	(-- addr)	P/F bit transmitted frame
K*	(-- addr)	Frame window size
RC*	(-- addr)	Retransmission Counter
N200*	(-- addr)	Maximum number of retransmissions
RC201*	(-- addr)	Retransmission Counter for TEI identity check
N201*	(-- addr)	Maximum number of octets in a received I frame information field
N201-TX*	(-- addr)	Maximum number of octets in an outgoing I frame information field
RC-ML1*	(-- addr)	Retransmission Counter for ID requests
N202*	(-- addr)	Maximum value for RC-ML1*
RC20*	(-- addr)	Retransmission Counter for XID procedure

4.5.2. RECEIVE VARIABLES

The following tables lists most of the Layer 2 receive variables. Note that the values are **only** valid if **EVENT-TYPE** contains the corresponding event. E.g. **FRAME#** variables are pertaining to the actual received frame only if **EVENT-TYPE** contains **FRAME#**. Checking the variables is only considered to be useful in a Test Manager environment, care must be taken when changing the values manually.

EVENT-TYPE (-- addr) Contains the type of the current event:

FRAME#	– A frame was received
LAYER1#	– Layer 1 event
FUNCTION-KEY#	– A function key was pressed
STRING-ENTRY#	– Keyboard input
TIME-OUT#	– A L2 emulation timer has expired
L1-TIME-OUT#	– A L1 timer has expired
TM-TIME-OUT#	– A Testscript timer has expired
TM-IPC#	– Test Manager IPC event
UNKNOWN-EVENT#	– Event not known by Layer 2

4.5.2.1. FRAME# EVENT VARIABLES

The following variables are pertaining to the last received frame if **EVENT-TYPE** contains **FRAME#**. The type of frame can be determined by checking the variable **FRAME-ID** (see below).

L2-POINTER	(-- addr)	Pointer to the first byte of the received frame
L2-LENGTH	(-- addr)	Length of the received frame
L3-POINTER	(-- addr)	Pointer to the first byte of the received packet
L3-LENGTH	(-- addr)	Length of the received packet
PORT-ID	(-- addr)	Identifier of receive direction (least significant 8 bits only), contains following values: TO_DTE_RX – Received from user side TO_DCE_RX – Received from network side
BLOCK-COUNT	(-- addr)	Sequential block count for received frame
START-TIME	(-- addr)	48 bit timestamp for the start-of-frame time
END-TIME	(-- addr)	48 bit timestamp for the end-of-frame time
MADDR1*	(-- addr)	First octet of address field
MADDR2*	(-- addr)	Second octet of address field

MCONTROL1*	(-- addr)	First octet of control field
MCONTROL2*	(-- addr)	Second octet of control field
MSAPI*	(-- addr)	Received SAPI
MTEI*	(-- addr)	Received TEI
CMND/RESP*	(-- addr)	Received frame COMMAND(=0)/RESPONSE (=1) indicator
MP/F-BIT*	(-- addr)	Value of POLL/FINAL bit
MNR*	(-- addr)	Received N(R)
MNS*	(-- addr)	Received N(S)
PIGGY*	(-- addr)	Piggybacked acknowledgement indicator (=1/=0). PIGGY* = 1 => the received I frame also acknowledged another I frame. (see also Test Manager 4.11 and 4.12)
FRMR-CNTL1*	(-- addr)	Frame Reject Control Field, Octet 1
FRMR-CNTL2*	(-- addr)	Frame Reject Control Field, Octet 2
FRMR-VR*	(-- addr)	V(R) in FRMR
FRMR-VS*	(-- addr)	V(S) in FRMR
FRMR-C/R-BIT*	(-- addr)	C/R in FRMR
FRMRW*	(-- addr)	W Bit in FRMR
FRMRX*	(-- addr)	X Bit in FRMR
FRMRY*	(-- addr)	Y Bit in FRMR
FRMRZ*	(-- addr)	Z Bit in FRMR
XID-TX-N201*	(-- addr)	Received peer N201-TX* value
XID-RX-N201*	(-- addr)	Received peer N201* value
XID-K*	(-- addr)	Received peer K* value
XID-T200*	(-- addr)	Received peer T200* value

FRAME-ID (-- addr) Contains frame identifier:

R#I	- Information frame
R#RR	- Receive Ready frame
R#RNR	- Receive Not Ready frame
R#REJ	- REJect frame
R#SABME	- Set Asynchronous Balanced Mode Extended frame
R#DISC	- DISConnect frame
R#UA	- Unnumbered Acknowledgement frame
R#DM	- Disconnected Mode frame
R#FRMR	- FRaMe Reject frame
R#UI	- Unnumbered Information frame
R#INVFRM	- INValid FRaMe
R-XID#	- Exchange Identity (XID) frame

If FRAME-ID contains R#UI, the type of UI frame can be determined by checking the value of the variable UI-TYPE*; if FRAME-ID contains R#INVFRM, the reason can be determined by checking the value of the variable DATA-STATUS (see below).

UI-TYPE* (-- addr) Contains the type of UI frame (valid only if FRAME-ID = R#UI):

DL-UNIT-DATA#	- Unnumbered Info (SAPI<>63)
MDL-UNIT-DATA#	- Management frame (SAPI=63 and TEI<>127)
TEI-ASSIGN#	- TEI management frame (SAPI=63 and TEI=127)

If UI-TYPE* contains TEI-ASSIGN#, the following variables are containing more detailed information about the received assignment frame:

MANAGEMENT-ID*	- Management entity identifier
RI*	- Reference number
MESSAGE-TYPE*	- Message type
AI*	- Action indicator (there can be more than one Action indicator in some TEI management frames: AI* contains the value of the first Ai)
#AI*	-The number of Action indicators in a TEI management frame

The next word gives all (max 8) Ai values from the TEI management message:

AI[] **(sequence number of Ai -- Ai or -1)**
The word leaves the value of the Ai if the given sequence number was legal, otherwise there will be -1 on the stack. (Example to get the second Ai in a TEI management message : 2 AI[])

Note: 1 AI[] = AI* @

The Message type contains one of the following constant values:

ID-REQ#	- Identity request	(U->N)
ID-ASSGN#	- Identity assigned	(N->U)
ID-DENIED#	- Identity denied	(N->U)
ID-CHK-REQ#	- Identity check request	(N->U)
ID-CHK-RESP#	- Identity check response	(U->N)
ID-REMOVE#	- Identity remove	(N->U)
ID-VERIFY#	- Identity verify	(U->N)

DATA-STATUS (-- addr) Received frame status indication

0x40 -> no error

Bit 0 = 1 -> Wrong Extension Bit
Bit 1 = 1 -> Aborted Frame
Bit 2 = 1 -> Receiver Overrun
Bit 3 = 1 -> Frame too short
Bit 4 = 1 -> CRC Error occured
Bit 7 = 1 -> XID Frame unknown Parameter Code
Bit 8 = 1 -> Invalid Frame Control byte value
Bit 9 = 1 -> Invalid Address byte
Bit 10 = 1 -> Wrong Size Supervisor/Unnumbered Frame
Bit 11 = 1 -> FRMR Frame Info Field Error

4.5.2.2. LAYER1# EVENT VARIABLES

If **EVENT-TYPE** contains **LAYER1#**, the type of the L.1 event can be determined by checking the variable **L1-ID*** , which then contains one of the following constants:

Basic Access:

R#ACTIVATE	- Layer 1 activated
R#DEACTIVATE	- Layer 1 deactivated
R#RI2ERR	- Info 2 received in unexpected state
R#LSTFRM	- Lost framing
R#RECOVERY	- Recovery for reported error
R#UNDEFINED	- Undefined error

Primary Rate Access :

R#REDALM	- Red alarm
R#YELALM	- Yellow alarm
R#LSTSIGL	- Lost of Signal Level
R#LSTPLOCK	- Lost of physical link lock
R#NORMAL	- Layer 1 back to normal
R#BPVL	- Bipolar violation
R#CRCR	- CRC error
R#MERR	- Multiframe error
R#FERR	- Frame error
R#CEPTSPEC	- CEPT special error

4.5.2.3. FUNCTION-KEY# , STRING-ENTRY# EVENTS

If EVENT-TYPE contains FUNCTION-KEY#, the key number will be in the variable **KEY-NUMBER**. See the K1197 FORTH Programming Language Manual for possible values of **KEY-NUMBER**. If EVENT-TYPE contains STRING-ENTRY#, either text or a number has been entered by the user (Testscenarios are able to extract the entered string or number, see the ISDN TEST MANAGER Manual).

4.5.2.4. TIME-OUT# , L1-TIME-OUT# AND TM-TIME-OUT# EVENTS

When EVENT-TYPE contains TIME-OUT# or L1-TIME-OUT#, the timer identification will be in the variable **TIMER-NUMBER**. When EVENT-TYPE contains TM-TIME-OUT#, a Testscenario can determine the timer identification (See the ISDN TEST MANAGER Manual).

4.5.2.5. TM-IPC# EVENTS

When EVENT-TYPE contains TM-IPC#, a Testscenario can determine the message and it's sender (see the ISDN TEST MANAGER Manual).

5. MISCELLANEOUS NOTES

5.1. "OUT OF BUFFERS" MESSAGE

This message is displayed when the pointer to the available Send Buffers is = 0. It is a normal result if attempting to send a large number of I frames. However, if it occurs at an unexpected time, such as right after starting the Layer 2 running, or establishing a data link, it is not correct, and indicates that the pointer has been inadvertently set to 0. To correct this condition, simply re-initialize the Emulation to either the User or Network Emulation. (USER_EMUL or NTWK_EMUL command, besides re-initializing the Send Buffers, it also initializes all the other Layer 2 variables).

5.2. XID PROCEDURES

The XID procedures are implemented in accordance with CCITT Recommendation Q.921 Blue Book, APPENDIX IV. However, there are some shortcomings in this description:

1. There is no description of the "negotiation" aspect of the procedure – i.e. what does a Layer 2 do with the XID parameters that it receives in an XID frame?
   The approach used in this implementation is:

   a. If an XID Command is received, the Layer 2 compares the received values with its internal preferred values. For N201-TX, N201-RX, and K, it adopts the lower values for its subsequent use; and for T200 it adopts the higher value for its subsequent use.

   b. The resulting XID Response frame that it sends contains its new Layer 2 values – i.e. the values just adopted.

   c. If an XID Response is received, the Layer 2 adopts the values exactly as received (except for the K value – see 5.2 (b) below), for its subsequent use, since the side that sent the Response (as a result of having received an XID Command) has already adopted values that are compatible with both sides.

2. There is no provision for both sides of the data link to negotiate different window sizes (only K-TX (transmit) is allowed in the XID message).
   The approach used in this implementation is to allow both sides to have their own unique K values:

   a. The Layer 2 really wants the peer Layer 2 to adopt a K that will keep the peer from sending too many I frames (which may overflow the Receive Buffer causing this Layer 2 to loose some frames). The objective is to reduce, or prevent Reject situations. To achieve this, the value that is placed in the K-TX parameter field in the outgoing XID frame is actually what it wishes the peer side to use for its K value – in this implementation's case, the K-RX variable which reflects the size of the Receive Buffer. Note that the K value that this side wants to use is not sent.

   b. When a frame is received, the Layer 2 adopts the lower of the received K value and its own K value.

3. No options are described for how many of the 4 parameters must be present in the XID frame. Hence, any XID frame that is not 3 or 21 bytes long will be considered to have an incorrect length.When an XID Command is sent via the State Machine, a full length frame is always sent. A short XID frame, both response and command, can be sent **outside** of the State Machine.

4. The procedure to be followed if an XID frame has one or more incorrect control codes is not described. Hence, the frame is considered to be totally invalid if any incorrect code is detected.

5. The procedure to be followed if an XID frame is received with either P or F=1 is not described. This implementation treats it as an error condition by notifying the ML or ASP and then discards it.

6. The procedure to be followed if an unsolicited XID response frame (F=0) is received, is not described – i.e. when this side has NOT sent an XID command (P=0) earlier. This implementation simply ignores it.

7. The procedures concerning how the Network changes its T201 duration when different T200 values are acquired for the various data links on one Basic Access are not described. Hence, only one value of T201 is used to time the responses to this request. To accommodate the slowest terminal (as indicated by the duration of its T200), T201 is made equal to the longest T200 duration of any of the terminals that have TEI's assigned to them. Thus, part of the XID procedure includes comparing the T200 just negotiated with the current T201 value, and making T201 equal to the longer of these durations. (Although this is also done on the USER Emulation as well, T201 of course is not used, so its value is irrelevant.)

8. The procedures concerning how a Terminal changes its T202 duration are not described. In this user implementation, T202 is made equal to twice the value of the T200 duration just negotiated.

5.3. DISABLING THE XID PROCEDURES

Because data link parameter initialization by using XID frames is implementation dependent (that is, optional), the Layer 2 State Machine's XID procedures can be disabled so as to ignore, and be totally transparent to, any incoming XID frames. However, the Monitor will still show the receipt of XID frames. Also, if the XID procedure is disabled, it will still be possible to execute all of the several XID commands which cause XID frames to be sent outside of the State Machine (Section 4.1.2.).

XID_ON	(--)	Enables the XID procedures
XID_OFF	(--)	Disables the XID procedures (default value)

5.4. OVERLOADING TRANSMIT & RECEIVE BUFFERS

It is possible to overload the Receiver Buffer (i.e. Layer 1) by the peer Layer 2 if it transmits a large number of I frames (that have been previously queued) and has a large window. When the buffer overflows, the incoming frames are totally lost, which eventually results in a Reject condition. This effect becomes more severe if the receiving side is displaying a lot of information on the screen, which slows down the speed at which it can process the received frames.

If it is desired to send a high volume of data, it is recommended that all screen prints be disabled. This can be done by disabling the TE, TX, and RX display, and by placing the Monitor in the REP_OFF mode (which inhibits the real time display of the external events, but does capture them in the buffer for subsequent viewing).

5.5. THE USE OF FRMR FRAMES

In Q.921 (3.6.11) it is described that in some certain cases an FRMR frame can be received. However, the SDL diagrams in ANNEX B and the state transition tables in ANNEX D never send an FRMR. In this implementation the State machine can always receive an FRMR but it never sends it.

The frame FRMR can be sent outside of the State machine control by using the commands **S_FRMR** and **SEND_FRMR**. **S_FRMR** sends the frame always with the last defined (by the user or the state machine) values of V(R), V(S), W, X, Y, Z etc. **SEND_FRMR** is used together with an MDL ERROR CODE. Codes J, L, N and O (see 4.3.1) describe the same error conditions as which are defined in Q.921 3.6.11 for FRMR frame as valid combinations of W, X, Y and Z bits. If **SEND_FRMR** is tried to be used with other codes, nothing happens.
Example:

```
ERR#J SEND_FRMR
```

sends an FRMR with W=0, X=0, Y=0 and Z=1 which indicates that there was an N(R) error.

5.6. TEI MANAGEMENT PROCEDURES

1. According to Q.921 there can be **more than one Action indicator** (Ai) in a TEI management message. However, it is not clearly defined what is the correct or possible amount of Ai's for each message type. In this implementation there must be exactly one Action indicator in messages *Identity request*, *Identity assigned* and *Identity denied*, or otherwise, the frame is considered to be invalid and thus ignored. In other TEI management message types any number (>0) of Ai's is allowed and also understood by the State machine.

   In this implementation the monitor does not store (in for Ai's specified variables) and display (in FRM_COMP mode) more than eigth Action indicators (from the beginning). The emulation sends as maximum six Ai's in one message.

2. The network emulation starts the **TEI check procedure** by sending an *Identity check request* message when an MDL-ERROR-INDICATION with an error code C, D, G or H (see 4.3.1) occurs or when it receives from a TE an *Identity verify* message. The Ai is set to the respective Ai value. If the reason to initiate the Identity check procedure was an MDL error, there is only one Ai in a message. In case of receiving an *Identity verify* message all action indicators from the received message are included in the *Identity check request* message.

   If another stimulus to start the TEI check procedure comes when there is already one going on, the existing check will be interrupted (and never finished) if
   - the new request for the procedure has a TEI/Ai which differs from the previous one and
   - the previous procedure was not already for all TEIs (Ai=127 in the *Identity check request*).

   In these cases a new TEI check procedure will be started with Ai=127 (this is not described in the Q.921 - actually, Q.921 does not say anything about a possible 'multiple TEI check procedure' of this kind). When the NT then receives the first response with the Ai of the previous check, it ignores it (because it is most propably a response to the already terminated check).

3. When an **MDL-ERROR-INDICATION with error codes C, D, G or H** is created, Q.921 gives the possibility to start the TEI check procedure (NT) or TEI identity verify procedure (TE) or just simply remove the TEI value (in case of NT by using the TEI removal procedure).

   In the above mentioned cases in this implementation the user emulation starts always the TEI identity verify procedure. The reaction of the NT emulation is as well always the TEI check procedure. (Of course in some cases these procedures end with TEI removal.) If the equipment, to which the K1197 is connected, does not support this feature, it will not cause any problems since the TEI identity verify procedure is designed to handle also this possibility.

4. When a **TE receives an Identity remove** message, according to Q.921 "Further action to be taken shall be either initiation of automatic TEI assignment for a new TEI value or notification to the equipment user for the need for corrective action". If the TEI to be removed was 0-63, a display message takes care of the above mentioned action. If the TEI was 64-126 no action is taken at the moment.

5. What to do with **erroneous TEI management messages**? When decoding a TEI management message, the following things are checked and possible errors are displayed (erroneous messages are ignored by the state machine):
   - Management entity identifier shall be (HEX) 0F.
   - The validity of the message type is checked.
   - The direction (TE->NT ; NT->TE) is check for each message type.
   - The value of the action indicators has to be as described in TABLE 8/Q.921. This is done separately for each message type.
   - The amount of Ai's
     - only one Ai is allowed in
       - *Identity request*
       - *Identity assigned*
       - *Identity denied*
     - any number of Ai's is allowed in
       - *Identity check request*
       - *Identity check response*
       - *Identity remove*
       - *Identity verify*.
   - The correctness of each action indicator extension bit.

   The reference number (Ri) values are **not** checked.

6. LAYER 3 SIMULATION

6.1. INTRODUCTION

The K1197 ISDN D-Channel Layer 3 Simulator is the name given to an environment that permits the user to easily create correct or incorrect messages in any sequence. These sequences can be controlled by received events, keyboard commands, timers, other programs, etc. Simulation is different to Emulation. Emulation provides a full and correct implementation of either the network side or the user side of the protocol. It does not send any incorrect messages.

This section describes the Simulation Environment of the ISDN Layer 3 software. The Simulation Environment provides functions (FORTH commands) which, when used in conjunction with the Test Manager, give a very powerful and flexible base for simulation of ISDN D-Channel signalling. This section presents an overview of how words from various software modules can be used to provide ISDN simulation. Detailed information is given only for those FORTH commands defined specifically for simulation. However references are given to further information about other FORTH commands. Note that all the FORTH commands described herein can be used as command inputs at any time while running the ISDN monitor/simulation software.

The simulation of the D-Channel Layer 3 protocol may be implemented in various degrees of complexity, depending on the requirements and experience of the user. This Simulation Environment provides a high level of programmer support for the novice user, particularly in the Message Building System where minimal protocol knowledge and programming experience are required to produce comprehensive tests. All simulation is based on the state machine concept as described in the Test Manager (section 6.4).

A basic understanding of the FORTH environment and language is necessary to use the simulation features effectively.

6.2. FUNCTIONAL GROUPS

The Layer 3 ISDN simulation Environment consists of the seven logical modules described below and shown in Figure 6-1. The first six are Test Manager support functions.

1. Main FORTH Dictionary (§ 6.3.1)	Extendable programming environment.
2. Layer 2 Emulation (§ 6.3.2)	Service over which the Layer 3 Simulator conveys Information between Layer 3 entities across the ISDN user-network interface using the D-Channel.
3. Layer 3 Decoder (§ 6.3.3)	Disassembly of Layer 3 messages.
4. Layer 3 Analyzer (§ 6.3.4)	Selection, display and storage of messages.
5. Message Building System (§ 6.3.5)	Assembly, storage, and retrieval of Layer 3 messages.
6. Layer 3 Management (§ 6.3.6)	Storage and access to information dealing with call connections.
7. Test Manager (§ 6.4)	A formalized method for implementing a state machine. Test Manager sequences may be stored and retrieved from disk files.

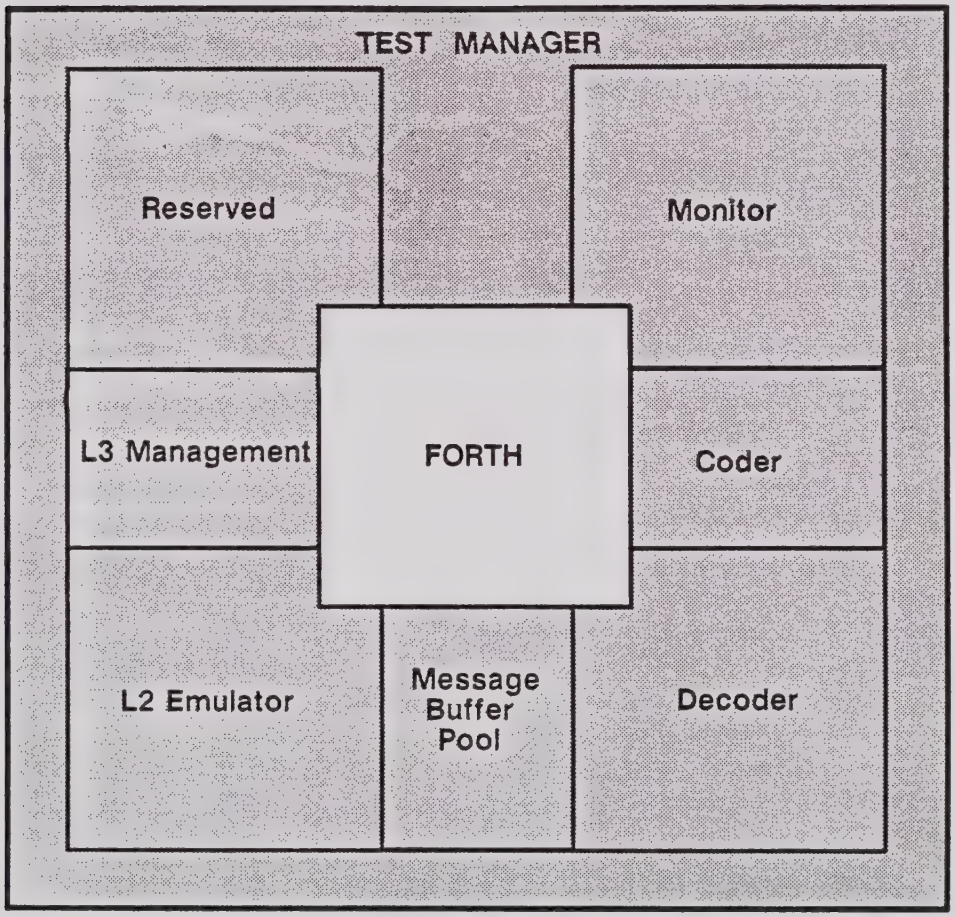


Figure 6-1: Schematic of Function Groups used for Simulation.

It can be seen that the Test Manager plays an important role in ISDN simulation. The FORTH commands in each module which are relevant for simulation are defined as the interface of that module to the Test Manager. In addition the main FORTH dictionary is accessible from the individual modules and also from the Test Manager.

6.3. TEST MANAGER SUPPORT FUNCTIONS

The following sections outline the features in other software modules which can be utilized by the Test Manager to perform ISDN Layer 3 simulation.

6.3.1. Main FORTH Dictionary

An extensive FORTH vocabulary containing words (commands) from FORTH-79, Fig-FORTH, and 68K FORTH, plus enhancements from Siemens FORTH is provided as the programming environment. There are no restrictions on the FORTH commands which can be accessed from the Test Manager. Consequently it is possible to use these commands to create new (user defined) FORTH words for simulation purposes.

6.3.2. Layer 2 Emulation

The Layer 2 Emulation provides the LAPD data link functions over which the Layer 3 simulator conveys information to the IUT (Implementation Under Test). Using the Layer 2 service primitives, the simulator can establish/release one or more data link connections on a D-channel, and transfer Layer 3 information via point-to-point data links or via broadcast data links. Furthermore, miscellaneous commands are provided to allow many Layer 2 internal variables to be modified to directly control the Layer 2 functions, such as the Data Link Connection Identifier (DLCI) and send/receive sequence number NS/NR.

Currently, when the K1197 is emulating Terminal Equipment, it supports 8 multiple TEI assignments. Each individual data link supports one Service Access Point. The following commands are used to support data link connections. (The information in parentheses indicates the data required on the stack and the data that is returned to the stack). The first four commands affect the data link in use for the current connection (see § 6.3.6).

DL_ESTABLISH	(--)
DL_RELEASE	(--)
DL_DATA_REQ	(data addr / data length --)
DL_UNIT_DATA_REQ	(data addr / data length --)
MDL_ALLOCATE	(SAPI -- CES / 1 or 0)
MDL_DEALLOCATE	(CES --)
MDL_SELECT	(CES / SAPI --)
MDL_EXISTED	(SAPI -- CES / 1 or 0)
MDL_INUSE	(CES / SAPI -- number of users)
MDL_REMOVE	(--)
MDL_CURRENT	(-- CES / SAPI)

DL_ESTABLISH and DL_RELEASE are used to establish and release a Layer 2 connection, while DL_DATA_REQ and DL_UNIT_DATA_REQ are used to send data using acknowledged and unacknowledged operation respectively.

MDL_ALLOCATE allocates a data link controller and sets its SAPI to the value given on the stack. If successful, the Connection Endpoint Suffix is returned together with a success flag of 1. If not successful, only a failure flag of 0 is returned. MDL_DEALLOCATE releases the data link for the given CES and resets the SAPI and TEI. MDL_SELECT searches for a data link associated with the given Connection Endpoint Identifier, and sets it to be the currently active link. If the given CEI does not exist, the current data link is unchanged. MDL_EXISTED determines if any data link is associated with the given SAPI value and active. If so, the Connection Endpoint Suffix of that data link is returned together with a success flag. MDL_INUSE returns the number of connections supported by the data link with the given SAPI/CES values, while MDL_REMOVE removes the association between the current link's TEI value and the connection endpoint. MDL_CURRENT returns the CEI of the current active link. (See also the LAYER 2 EMULATION (section 4) for further information on Layer 2 events).

6.3.3. Layer 3 Decoder

The Layer 3 Decoder disassembles a received signalling message. All protocol information, such as the message type and the contents of each field of each information element, are stored in Layer 3 internal variables. The simulator can read these variables and determine the corresponding action. The ISDN TEST MANAGER MANUAL contains details about these variables and how to access them.

6.3.4. ISDN Monitor

The ISDN Monitor provides for the comprehensive monitoring and display of ISDN D-channel protocol information at Layers 2 and 3 (plus some Layer 1 event information). Simultaneous display of the signalling information for both directions of a link is possible. In addition it is possible to analyze and/or record only those events selected by the use of triggers and filters. The information received may be reported (decoded and displayed), stored in a Capture RAM and/or recorded on Disk for analysis or use at a later time. The Monitor is completely *passive*. Consequently it does not affect the signalling information that it monitors.

The Monitor provides a wide range of formats for the display of ISDN Layer 1, Layer 2 and Layer 3 information. Each layer has independent display format control so that the user can select the combination of display formats desired.

All the functions of the ISDN D-Channel Monitor may be utilized during simulation. It is possible to access, from within the Test Manager, all FORTH commands defined in the Command, Programmable Triggers, and Programmable Filters sections of the ISDN Monitor User Manual.

6.3.5. Message Building System

The Message Building System consists of a Coder for constructing Layer 3 signalling messages, a menu-based interface to the coder allowing fast and user-friendly construction of such messages, and a Message Buffer Pool for the storage of messages so constructed.

The Layer 3 Coder provides a programming interface for the setting of information element parameters and the creation of information elements and messages. Messages which are created can be either stored for later use or transmitted immediately.

The information elements are created using the Information Element building commands. These commands automatically access the relevant parameters related to each information element and extract the data required to build the information element octet by octet. The contents of each parameter and the structure of each information element (i.e. which octets are present) are definable by the user and can be easily changed using either the menu driven Message Building System or FORTH commands.

Messages are built by specifying the required message type and the Information Elements to be included and using the programming interface commands to combine these into a message. As with the information elements, the user can change the structure and contents of the messages being created. The user can use either the menu driven Message Building System or FORTH commands to define which information elements will be included in a message and to define in what order they will appear within a message.

When the message has been built, it can either be transmitted immediately in an I frame or a UI frame, it can be stored in a buffer for later use, or it may be stored as part of a Message Buffer Pool in a disk file for use in subsequent testing sessions.

The Message Buffer Management System is used to manage a collection of Layer 3 messages for simulation purposes. It contains 60 message buffers, each of which contains a name field (used to identify the message), a comment field, and a data field which contains the message itself. Both the name fields and the comment fields are stored in the pool as normal FORTH strings, while the data field uses the first two bytes to denote the message length, and the remaining space for the message itself.

The length of a message which may be stored in a single data field is 130 octets. However, an extension mechanism exists whereby two adjacent entries may be used to store an extended message (up to 262 bytes). The operation of this mechanism is fully automatic, and will occur when a message is stored which is too large to fit in a single entry.

The programming interface to the Message Buffer Management System enables the user to access specific, predefined Layer 3 messages from the Test Manager and to use them in test scenarios.

For a detailed description and examples of the programming interface commands of the Layer 3 Coder and the Message Buffer Management System, see the section 3 and 6.6.

6.3.6. Layer 3 Connection Management

The Layer 3 Simulation supports the signalling procedures for up to 8 simultaneous connections (CNs). The Connection Management Module is responsible for maintaining 8 identical data structures, which contain data for each of these connections. At any one time, only one of these connections (and hence one of the data structures) is "active", and any reading or writing of connection-related data is done using the active data structure (see Figure 6-2).

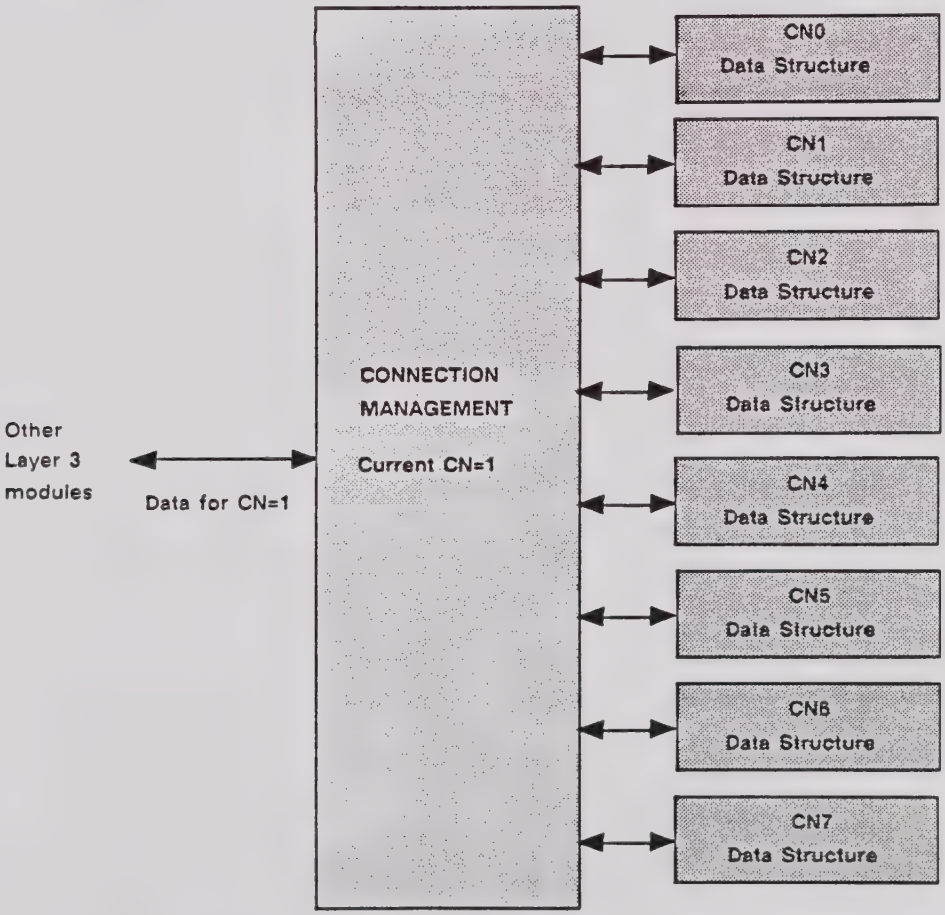


Figure 6-2: Multiplexing of Connection related data.

When an event occurs in the simulation environment, it is examined to determine which of the connections it is destined for, e.g. when a Layer 3 message is received, the call reference is compared against those of all connections. If a match is made, the data structure for that connection is made "active", and the data it contains is used in generating the simulator's response to the event. For example the send commands from the MBS Monitor use the current settings of CR value, destination, for generating the Message header. Each Testmanager is using the corresponding connection data structure. (TM0 uses CN0 .... TM7 uses CN7)

The module interface consists of several FORTH commands which are described in the following sections. In general, these commands set or return a particular data item (e.g. a B channel number, the local address) for the "active" connection.

Each connection is represented by a number (0–7). The currently active connection can be determined by the FORTH command

```
?CN   ( -- currently active CN )
```

The active connection can be set by any of the following commands

```
=CN   ( CN -- )
CN0   ( -- )
CN1   ( -- )
CN2   ( -- )
CN3   ( -- )
CN4   ( -- )
CN5   ( -- )
CN6   ( -- )
CN7   ( -- )
```

The constants #CN0, #CN1, up to #CN7 are defined for use with =CN. For example, the following sequence

```
#CN4 =CN
```

will make connection number 4 the current connection.

NOTE: that the LAYER 3 Simulation Enviroments the commands =TM, TM0 .. TM7 are performing the same actions as =CN, CN0 .. CN7. (See also the ISDN TEST MANAGER Manual)

6.3.6.1. Connection-related Data Management

For each CN, the Connection Management Module maintains the following data:

- the calling number and subaddress
- the called number and subaddress
- the channel type in use for the connection (B or D)
- the channel number in use for the connection (only valid when channel type = B)
- the national and international protocol discriminators
- the service access point identifier (SAPI)
- the connection endpoint suffix (CES)
- a flag representing the call reference flag
- the call reference value
- a set of all the information element parameters used in coding Layer 3 messages
- a set of pre-coded IEs

To access single parameters for a particular connection, the following FORTH commands are defined.

```
=CN_CALLING_NUM          ( calling number address -- )
?CN_CALLING_NUM          ( -- calling number address )
=CN_CALLED_NUM           ( called number address -- )
?CN_CALLED_NUM           ( -- called number address )
=CN_CALLING_SAD          ( calling subaddr address -- )
?CN_CALLING_SAD          ( -- calling subaddr address )
=CN_CALLED_SAD           ( called subaddr address -- )
?CN_CALLED_SAD           ( -- called subaddr address )
=CN_CHANNEL_TYPE         ( channel type -- )
?CN_CHANNEL_TYPE         ( -- channel type )
=CN_CHANNEL_NUM          ( B-channel number -- )
?CN_CHANNEL_NUM          ( -- B-channel number )
=CN_PD                   ( pd value -- )
?CN_PD                   ( -- pd value )
=CN_CEI                  ( CES / SAPI -- )
?CN_CEI                  ( -- CES / SAPI)
```

In each case, the parameter which is set or returned pertains to the currently active CN . The constants #B_CHAN and #D_CHAN are defined for use with =CN_CHANNEL_TYPE and ?CN_CHANNEL_TYPE. For example

```
#D_CHAN   =CN_CHANNEL_TYPE
```

is used to indicate that the channel being used for the current connection is the D-channel. (In this case, the channel number for the connection is meaningless).

Each Layer 3 connection has associated with it a call reference, which is used in the header of Layer 3 signalling messages for that CN. The length of the call reference is the same for all CNs, not just the current one. The following commands are used to set and return the length of the call reference field in octets.

=CR_LENGTH	**(call ref length --)**
?CR_LENGTH	**(-- call ref length)**

The call reference value for each CN and flag may be set or accessed by the following commands.

=CN_CR_SOURCE	**(call ref source --)**
?CN_CR_SOURCE	**(-- call ref source)**
=CN_CR_VALUE	**(call ref value --)**
?CN_CR_VALUE	**(-- call ref value)**

The call reference source is a boolean variable which set to the constant **#OUTGOING** if the connection was initiated by the K1197, and **#INCOMING** if it was initiated by the device under test. This field is used in coding the call reference flag in outgoing Layer 3 messages, and interpreting the flag in incoming messages.

6.3.6.2. Call Reference Management

The following commands are also used to manipulate call references in the Connection Management Module.

CR_VAL_MAX	**(-- callref)**
SELECT_CR	**(-- callref)**
CN_INIT	**(direction / SAPI --)**
CLEAR_CR	**(--)**

CR_VAL_MAX returns the maximum possible call reference value based on the current call reference length. SELECT_CR returns a call reference value between 1 and CR_VAL_MAX which is not currently in use for an outgoing call. 0 is returned if none are available. CN_INIT is used to set the SAPI and the call reference value and flag for the current CN. The direction of the CN may be specified using the constants **#INCOMING** and #OUTGOING. If **#INCOMING** is used the call reference for this CN is set automatically to that of the last decoded message, otherwise it is set by using SELECT_CR. In addition, the protocol discriminators (national and international), the default IE field values are set and any possible IE is generated for the current CN. CLEAR_CR is used to set the call reference value for the active CN to **#UNUSED**, which indicates that the call reference has been released.

6.3.6.3. Information Element Management

The Connection Management Module maintains a set of all the different parameters used to construct information elements for each connection. These are used to store decoded parameters from messages pertaining to that connection, as well as being used in the construction of outgoing messages. The base address of the area of memory where these parameters are stored for the active CN is returned by the command

```
?CN_IE_LIST
```

The following FORTH commands are defined to copy sets of these parameters.

COPY_PARAMS	**(srce CN / dest CN --)**
COPY_IE	**(srce CN / dest CN / IE index --)**

COPY_PARAMS is used to copy all the parameters for all IEs from one CN to another, while COPY_IE only copies those parameters related to a specified IE. Note that both these words may be used to copy parameter values from the decoder instead of a CN by specifying **#DECODER** as the copy source. For example, the sequence

```
#DECODER  #CN4 I#BEARER_CAP COPY_IE
```

will copy the parameters in the "Bearer Capability" IE of the last received message from the decoder into CN 4.

In addition, a set of buffers is kept for each CN for storing generated IEs. The following commands allow the user to manipulate these buffers and their contents.

INIT_IE_BUFFERS	**(--)**
LONG_BUFFER	**(IE index --)**
RESET_IE_MAP	**(--)**
RESET_ALL_IE_MAPS	**(--)**
?CN_IE_BUFFER	**(IE index -- addr of buffer or 0)**
COPY_BUFFERS	**(source index / dest index --)**
COPY_IE_BUF	**(CN srce / CN dest / IE index --)**

INIT_IE_BUFFERS is used to clear the contents of all the IE buffers for the active CN. There exists one buffer for each IE type. All IE buffers default to a 40 byte length but may be extended in length (122 bytes) by using the command LONG_BUFFER. For example, the sequence

```
I#BEARER_CAP LONG_BUFFER
```

will assign a long buffer to the "Bearer Capability" IE for the active CN. If an attempt is made to extend the buffers for more than 3 IEs, an error message is given and no action is performed.

All IEs may be reset to the smaller length for the active CN by using the command RESET_IE_MAP. The command RESET_ALL_IE_MAPS is used to do the same thing for all CNs.

The command ?CN_IE_BUFFER returns the address of the buffer for the specified IE for the active CN. 0 is returned if the specified IE identifier is unknown.

The command COPY_BUFFERS copies all the IE buffers from one CN to another. The command COPY_IE_BUF will copy only a single IE buffer. If #DECODER is specified as the source, COPY_IE_BUF will copy the complete IE from the last decoded message (if it was present in that message) to the destination CN. For example the sequence

```
#DECODER #CN7 I#BEARER_CAP COPY_IE_BUF
```

will copy the "Bearer Capability" IE, if it was present in the last decoded Layer 3 message, from the decoder to CN7.

6.3.6.4. B-Channel Management

A table is maintained by the Connection Management Module to manage B-channels. This table contains as many entries as there are B-channels supported at the interface, with each entry containing information to indicate whether a B-channel is currently allocated and also whether it is currently connected.

The following commands exist for accessing the B-channel table.

BC_INIT	(--)
BC_ALLOC	(bchan --)
BC_DISC	(bchan --)
BC_FREE	(bchan --)
BC_CONN	(bchan --)
?BC_CONN	(bchan -- flag)
?BC_ALLOC	(bchan -- flag)
?BC_SELECT	(-- bchan)
SET_INTERFACE	(interface --)

BC_INIT is used to reset the B-channel table, i.e. to mark all B-channels as being free. BC_ALLOC, BC_DISC, BC_FREE and BC_CONN are used to flag the specified B-channel as being allocated, disconnected, free or connected respectively. The distinction between a channel being allocated and connected is that an allocated channel may not yet be connected, but it prevents that channel being used by any other connection. Note that BC_FREE marks the B-channel as being free and disconnected, and BC_CONN marks the B-channel as being connected and allocated.

?BC_CONN and ?BC_ALLOC are used to determine the connection and allocation status of the specified B-channel, while ?BC_SELECT returns the number of a B-channel that is marked as being free. If all B-channels are in use, 0 is returned. SET_INTERFACE is used to set the number of B-channels, and clear the B-channel table. In addition, the number of octets used in coding call references is set, as this is also dependent on the interface type. The interface is specified by using the constants #PRIMARY (call reference length = 2 octets, number of B-channels = 30) and #BASIC (call reference length = 1 octet, number of B-channels = 2).

6.3.6.5. Timer Management

The following description refers to the CCITT defined timers. National variants are specified in the National Variant User Manual.

The Layer 3 simulation supports the use of 4 timers for each CN. Two of these are used to implement the CCITT (or other national variant) defined timers (listed below), while the remaining two are general purpose timers.

With the exception of T312 (started when the network side of the interface has sent a broadcast SETUP message and is awaiting a valid response), only one of the CCITT defined timers may be running at any one time for a particular connection. A single system timer can therefore be used to implement all these timers for that connection. Timer T312 and the two general purpose timers are each implemented with their own system timers, and may be running simultaneously with all the other timers in the one connection.

The CCITT and general purpose timer durations are maintained by the Connection Management Module for each CN. The timer names, their respective FORTH constants, and their default durations are described in the National Variant User Manual.

To manipulate these timers and their durations, the following FORTH commands are defined.

INIT_DEFAULT_TIMERS	**(--)**
=CN_TIMER_DUR	**(timer constant / time --)**
?CN_TIMER_DUR	**(timer constant -- time)**
CN_START_TIMER	**(timer constant --)**
CN_STOP_TIMER	**(timer constant --)**
?CN_TIMER	**(timer constant -- flag)**

INIT_DEFAULT_TIMERS is used to reset all timer durations to their default values as indicated in the table above for every CN. The commands =CN_TIMER_DUR and ?CN_TIMER_DUR are used to set and get, respectively, the timer duration for the specified timer for the active CN. In each case, the time is specified in tenths of a second. For example the following FORTH sequence

```
#T312 40 =CN_TIMER_DUR
```

sets timer T312 for the current connection to a duration of 4 seconds. The sequence

```
#USER2 ?CN_TIMER_DUR
```

leaves the duration (in tenths of a second) of the general purpose timer number 2 for the active CN on the stack.

The commands CN_START_TIMER and CN_STOP_TIMER are used to start and stop the specified timer for the active CN , while ?CN_TIMER is used to return a flag indicating whether the specified timer for the active CN has expired (non-zero indicates that the timer has expired).

6.3.6.6. Connection Multiplexer

In order to support multiple simultaneous connections, a layer 3 event multiplexer is provided. The connection multiplexer maintains a current connection for each Test Manager. Note that there is a strong relationship between CN's and Test Manager's when the connection multiplexer is active.

The following two commands allow the Layer 3 multiplexer to be switched on and off.

L3_MUX_ON	**(---)**
L3_MUX_OFF	**(---)**

When the Layer 3 multiplexer is switched off, the currently active Test Manager is executed immediately in response to an event. The Test Manager state is only changed by using the command NEW_STATE within the specified actions, and is independent of the input event.

When the Layer 3 multiplexer is switched on, any incoming event is examined to determine which connection, and therefore which Test Manager should handle it. The Test managers pertaining to these connections are then made active in their current states.

The following strategy is used for searching Test Managers pertaining to the event:

If the event was a received message, then the Test Manager is executed either for the first detected "specialized" CN or, if no specialized CN was found for the message, for the first detected "half specialized" CN or, if no half specialized CN was found for the message, for all "general" CN's.

This is done automatically by the connection multiplexer by comparing the parameters of the last received messages with those parameters that are set up for each CN.

A CN is called specialized when it is set up with particular values for :

- the call reference
- the CES (connection endpoint suffix)
- the SAPI (service access point identifier)
- the call direction
- the PD (protocol discriminator)

A CN is called half specialized when it is set up with particular values for :

- the SAPI (service access point identifier)
- the call direction
- the PD (protocol discriminator)

  and **either**
- #UNUSED for the call reference to indicate that all call references are valid for this connection

  or
- #UNUSED for the CES to indicate that all connection endpoint suffixes are valid for this connection

A CN is called general when it is set up with particular values for :

- the SAPI
- the call direction
- the PD

  and
- #UNUSED for the call reference to indicate that all call references are valid for this connection

  and
- #UNUSED for the CES to indicate that all connection endpoint suffixes are valid for this connection

STOP_SCANNING (---)

calling STOP_SCANNING from a Test Manager action will cause that no more scripts will be activated for the current event, even if they would match. This function is useful, when one particular Test Manager wants to execute the last event and prevent other Test Managers from receiving and executing the same event once more.

In case of timer events the Test Manager that started the particular timer is activated.

For keyboard events the currently active Test Manager (i.e. the Test Manager whose function key labels are visible) is activated. On the contrary, a supervisor control function is available to handle keyboard events only by TM0. The following two commands allow to switch on or off the supervisor control.

SVC_ON (---)
SVC_OFF (---)

The initialization of a connection must be done in state 0 to ensure the proper operation of the connection multiplexer. When the multiplex connections for a particular connection are not clear at the time of initialization, those conditions should be set to #UNUSED. In general, the Test Manager actions should make use of the Connection Management commands which only access or set data relevant to the currently active connection.

CN_INIT should be used to initialize the currently active connection.

CN_INIT (direction/SAPI ---)

initializes the multiplex conditions "direction"," call reference"," CES "and "SAPI" for the currently active connection.The direction of the connection may be specified using the constants **#INCOMING** or **#OUTGOING.**
If the direction is specified to be #INCOMING, the CES and call reference values are taken from the last received message.
If the direction is #OUTGOING, the CES is set to #UNUSED in NT mode or the current active link in TE mode. Additionally, a "free" call reference value is automatically selected and set for this connection.

The following testscript is an examle to use the connection multiplexer in the network emulation. Two testscenarios will be loaded into two different testmanagers: TM0 will handle the incoming calls and TM1 will handle the outgoing calls. A poolfile " POOLFILE.P" must exist where the neccessary messages are stored (see message building system, section 3 and 6.6 of this manual for detailed information). The example only shows the initialization of the testmanagers and the actions until the layer 3 is connected. Further actions might be implemented.

```
(-------------------------------------------------------)
( File :          LOAD.F                                )
( Project:        Loader file for testscript            )
(-------------------------------------------------------)

: LDTEST ( --- )
        GLOB_TCLR
        " POOLFILE.P" LOAD_MESSAGES
    TM0 " INCOMING.F" EXECF
    TM1 " OUTGOING.F" EXECF
    GLOB_TM_RUN
;
L3_MUX_ON
LDTEST                                  ( last line of the loader file ! )
                                        ( further commands will not be )
                                        ( executed ! )

(-------------------------------------------------------)
( File :          INCOMING.F                            )
( Project:        Test scenario for incoming calls      )
(-------------------------------------------------------)

TCLR
0 STATE_INIT{
                                        ( First Initialisation of L3-MUX )
    #INCOMING  =CN_CR_SOURCE            ( CR-flag INCOMING )
    #UNUSED    =CN_CR_VALUE             ( CR-value : every value matches )
    #UNUSED 0  =CN_CEI                  ( CES : every value matches )
                                        ( SAPI : only SAPI=0 matches )
    ?CN_INT_PD =CN_PD                   ( PD : only PD=INT-PD matches )
}STATE_INIT
0 STATE{
    M#SETUP 1 ?L3_VALID_MSG
    ACTION{
                                        ( Second Initialisation of L3MUX )
        #INCOMING 0 CN_INIT             ( CES and CR - value are taken from )
                                        ( the SETUP - msg, so they are made )
                                        ( " special" )
        " CALL_SENT" SEND_MESSAGE
        " CONN"      SEND_MESSAGE
        10 NEW_STATE                    ( Active state )
    }ACTION
}STATE

         ...

10 STATE{
         ...
}STATE
```

```
(-------------------------------------------------------)
( File :          OUTGOING.F                           )
( Project:        Test scenario for outgoing calls     )
(-------------------------------------------------------)

TCLR
0 STATE_INIT{
                                          ( First Initialisation of L3-MUX )
    #OUTGOING  =CN_CR_SOURCE              ( CR-flag OUTGOING )
    #UNUSED    =CN_CR_VALUE               ( CR-value : every value matches )
    #UNUSED 0  =CN_CEI                    ( CES : every value matches )
                                          ( SAPI : only SAPI=0 matches )
    ?CN_INT_PD =CN_PD                     ( PD : only PD=INT-PD matches )
}STATE_INIT
0 STATE{
    FK_CF1
    ACTION{
                                          ( Second Initialisation of L3MUX )
        #OUTGOING 0 CN_INIT               ( New CR value for this transaction )
                                          ( CR - value is made " special" )
        " SETUP" SEND_UNIT_MESSAGE
        #T303 CN_START_TIMER              ( Start timer T303 )
        6 NEW_STATE                       ( " CALL PRESENT" state )
    }ACTION
}STATE
6 STATE{                                  ( " CALL PRESENT" state )
    M#ALERT 1 ?L3_VALID_MSG
    ACTION{
        #T3AA CN_START_TIMER
    }ACTION
    M#CONN 1 ?L3_VALID_MSG
    ACTION{
        #T3AA CN_STOP_TIMER
        #T303 CN_STOP_TIMER               ( Stop timer T303 )
                                          ( Third initialisation of L3MUX )
        MDL_CURRENT =CN_CEI               ( CES is made "special", the value )
                                          ( is taken from the CONN - Msg )
                                          ( Now this TM listens just to : )
                                          ( CR - flag : OUTGOING )
                                          ( CR - value : as selected )
                                          ( CES : as selected )
                                          ( PD : INT-PD )
        " CONN_ACK" SEND_MESSAGE
        10 NEW_STATE                      ( active state )
    }ACTION
}STATE
        ...

10 STATE{
        ...
}STATE
```

6.3.7. B / D Channel - Communication

6.3.7.1. Introduction

To allow communication between D-channel and B-channel application packages a programming interface is needed. The interface described below provides functions for B-channel application packages to request physical connection / disconnection of a B-channel.

The interface does also provide functions to inform the D-channel software, that a B-channel application is present to accept an incoming call and that the B-channel software should be notified, when an incoming call was received.

Communication with B-channel applications can be implemented by using the Test Manager mechanism. The following Section gives an overview of service primitives that are provided to connect and disconnect a call. The use of the Test Manager mechanism in the B-channel application is optional. If not used, connection and disconnection responses will be done automatically.

6.3.7.2. Description of Service Primitives

The Connect / Disconnect service primitives are used in the Test Manager by the B- and D-channel applications as shown below (section 6.3.7.3).

B/D_CONN_REQ ..... (b-channel --)
B/D_CONN_RES ..... (b channel --)

After receipt of a SETUP message the D-channel side may inform the B-channel application by sending a **B/D_CONN_REQ** with a parameter indicating which B-channel should be used. If there is any B-channel-application which is able to accept the call, the B-channel application should accept the call by sending a **B/D_CONN_RES** to inform the D-channel application. When the B-channel application initiates a call, it will send a **B/D_CONN_REQ** which should be normally answered with a **B/D_CONN_RES** on the D-channel side, after a corresponding CONNECT message has been received. If the call could not be connected on the D-channel side, then a **B/D_DISC_REQ** should be sent.

B/D_DISC_REQ ..... (b-channel --)
B/D_DISC_RES ..... (b-channel --)

After receipt of a DISCONNECT message the D-channel side should send a **B/D_DISC_REQ** to the corresponding B-channel application. The request will then be confirmed by a **B/D_DISC_RES**. When the B-channel application clears the call a **?B/D_DISC_IND** will be true on the D-channel side. After receipt of a RELEASE - message the indication should be responded with a **B/D_DISC_RES.**

Connect/disconnect indications/confirmations can be recognized by using the following Test Manager event recognition words:

?B/D_CONN_IND ..... (-- flag)
?B/D_CONN_CNF ..... (-- flag)
?B/D_DISC_IND ..... (-- flag)
?B/D_DISC_CNF ..... (-- flag)

If a connect request will be unsuccessful, the application will give this information to the Test Manager. The Test Manager will get this information by using:

?B/D_DET_IND ..... (-- flag)

It is possible to pass or receive information to or from the B-channel side within a connect request.

=B/D_UPTR ..... (value --)
?B/D_UPTR ..... (-- value)
=B/D_CALLED ..... (called number adr --)
?B/D_CALLED ..... (-- called number adr)
=B/D_CALLING ..... (calling number adr --)
?B/D_CALLING ..... (-- calling number adr)

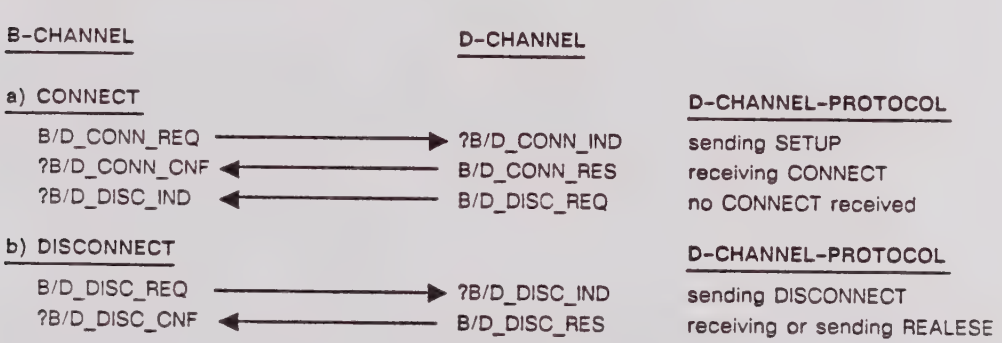
The value can be ie. a user defined pointer which points to a memory area filled with information.

These Test Manager events may be controlled by using a general purpose timer.

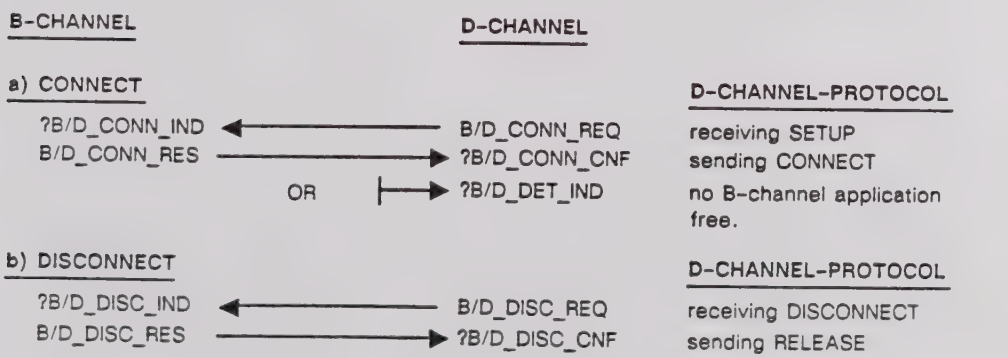
6.3.7.3. Communication Strategy

The B / D channel communication is implemented as a state machine using service primitives. Incoming and outgoing calls may be implemented as follows :

EXAMPLE 1) OUTGOING CALL



EXAMPLE 2) INCOMING CALL



6.4. TEST MANAGER

6.4.1. Introduction

The Test Manager provides a formalised (but very easy to learn) programming environment which is based on CCITT SDL logic (STATE -> EVENT -> ACTION -> NEWSTATE logic). Therefore once the test sequence is specified in SDL it is very easy to convert the specification into a sequence of FORTH commands. The Test Manager syntax provides a number of commands which allow the user to assemble the test scenario as a finite state machine (for more information, see the ISDN Test Manager User Manual).

It is possible to invoke the Layer 3 Test Manager from either the Analyzer or the Emulator. When invoked by the Analyzer, it serves as an analytical tool which allows the user to collect statistics and evaluate protocol procedures while monitoring live data or playing back captured or recorded data in selected data formats. When invoked by the Emulator, the Test Manager has all the analytical power described above, and the additional capability to send tailored messages as part of the test scenario for protocol and service testing.

6.4.2. Evaluation of Layer 3 Events

During simulation, there are many different types of events that can occur which have to be processed. These events include reception of a frame, timer expiry, keyboard entries, etc. Most of these events are already handled automatically by the Layer 2 Emulator software. However, the Layer 3 Simulator indicates a Layer 3 event on the reception of Layer 3 messages or expiry of Layer 3 related timers. The nature of these events are determined by user defined test scenarios. The following FORTH commands provide the user with the ability to determine the type and content of received Layer 3 events.

?L3_MSG	**(MSG index / ... / MSG index / n -- flag)**
?L3_IE_COUNT	**(IE index -- count)**
?L3_IE	**(IE index / n -- flag)**
?L3_MAND_IE	**(MSG index -- flag)**
?L3_OPT_IE	**(MSG index -- flag)**
?L3_UNEXP_IE	**(MSG index -- flag)**
?L3_OCTET	**(IE index / octet index -- flag)**
?L3_MATCH	**(string -- flag)**
?L3_IE_ORDER	**(-- flag)**
?L3_VALID_MSG	**(MSG index / ... / MSG index / n -- flag)**
?CN_TIMER	**(timer constant -- flag)**

These FORTH words are described in full in the "ISDN Test Manager Manual". For a description of ?CN_TIMER, see Section 6.3.6.5, "Timer Management"

6.4.3. Sending Messages

Layer 3 messages may be transmitted to line from the Simulator by several methods. These methods differ in the way the transmitted message is specified and the type of Layer 2 frame used to transport the message (ie. I – information, or UI – unnumbered information frames).

If the message to be sent is specified as a "raw" hexadecimal string, there are two methods of transmission. DL_DATA sends the message string in a Layer 2 I frame, while DL_DATA sends the message string in a Layer 2 UI frame. For example, the FORTH sequences

```
X" 0801C50D" DL_DATA
X" 0801C50D" DL_UNIT_DATA
```

will cause two SETUP ACKNOWLEDGE messages to be sent, the first using an I frame and the second using a UI frame.

These two commands can only send Layer 3 messages of up to 255 bytes. For longer messages, the commands DL_DATA_REQ and DL_UNIT_DATA_REQ must be used. These differ from the previous two commands in that they require the address of the start of the message and the number of bytes. For example, the FORTH sequences

```
X" 0801C50D" COUNT DL_DATA_REQ
X" 0801C50D" COUNT DL_UNIT_DATA_REQ
```

will cause the same two SETUP ACKNOWLEDGE messages to be sent as before, using an I frame and a UI frame respectively.

If the message to be sent it is stored in the Message Buffer Pool (section 6.3.5), it may be accessed by using the name field of the relevant Pool Entry. Two FORTH commands, SEND_MESSAGE and SEND_UNIT_MESSAGE, are defined to allow these stored messages to be transmitted. As an example, the FORTH sequences

```
" MESSAGE1" SEND_MESSAGE
" MESSAGE2" SEND_UNIT_MESSAGE
```

cause the data fields of the Pool Entries with the name fields of "MESSAGE1" and "MESSAGE2" to be sent via Layer 2 I and Layer 2 UI frames respectively. In each case, if the required name field cannot be found in the Message Buffer Pool an error message is displayed and nothing is transmitted.

In addition it is possible to construct and send a message in one FORTH command sequence with the words MESSAGE>, <SEND and <SEND_UNIT. A detailed description of how these words are used can be found in the "Program Building Method of Generating Messages" (section 6.7). As an example, the FORTH sequence

```
M#SETUP MESSAGE> I#BEARER_CAP I#CHAN_ID <SEND
```

will generate a SETUP message containing the BEARER CAPABILITY and CHANNEL ID information elements and transmit the message via a Layer 2 I frame. The <SEND_UNIT command performs the same function as <SEND , except a Layer 2 UI frame is used to transport the Layer 3 message.

Three commands are provided to set up the mode for the send commands SEND_MESSAGE, SEND_UNIT_MESSAGE, <SEND and <SEND_UNIT:

```
AUTO_HDR_UPDATE     ( { ON | OFF } -- )
AUTO_BROADCAST      ( { ON | OFF } -- )
AUTO_UI_SETUP       ( { ON | OFF } -- )
```

If AUTO_HDR_UPDATE is switched on the words SEND_MESSAGE and SEND_UNIT_MESSAGE will use the CR-flag and CR-value from the currently active or, if #UNUSED, from the last received message for sending the specified message. If switched off, CR-flag and CR-value are taken from the pool message (i.e. the "raw" pool message is sent). **Note :** <SEND and <SEND_UNIT always use the values defined for the current active CN-structure.

AUTO_BROADCAST controls the mode of sending UI-frames. If AUTO_BROADCAST is swiched on and the simulation software is AUTO_HDR_UPDATE configured as NT, SEND_UNIT_MESSAGE and <SEND_UNIT will transmit UI-frames as "broadcast".

AUTO_UI_SETUP controls the mode of automatic UI sending of SETUP messages on the NT side. If AUTO_UI_SETUP is swiched on and the simulation software is configured as NT, SEND_MESSAGE and <SEND will transmit a UI frame, if the message type is "SETUP". **Note:** This UI will be also transmitted as "broadcast" when if AUTO_BROADCAST is switched on.

6.5. LAYER 3 MANAGEMENT USER INTERFACE

The following commands provide the programming interface to the Layer 3 Management Module. (The information in parenthesis indicates the before and after status of the FORTH stack.)

6.5.1. Connection Management

=CN (CN --)
sets the currently active connection to CN.

CN0 ... CN7 (--)
selects connection number 0 - 7 as the currently active connection.

?CN (-- CN)
returns the currently active connection number.

6.5.2. Connection Parameter Access

=CN_CALLING_NUM
(calling number string --)
sets the calling number string for the currently selected connection.

?CN_CALLING_NUM
(-- calling number string)
returns the calling number string for the currently selected connection.

=CN_CALLED_NUM
(called number string --)
sets the called number string for the currently selected connection.

?CN_CALLED_NUM
(-- called number string)
returns the called number string for the currently selected connection.

=CN_CALLING_SAD
(calling subaddr string --)
sets the calling subaddress string for the currently selected connection.

?CN_CALLING_SAD
(-- calling subaddr string)
returns the calling subaddress string for the currently selected connection.

=CN_CALLED_SAD
(called subaddr string --)
sets the called subaddress string for the currently selected connection.

?CN_CALLED_SAD
(-- called subaddr string)
returns the calling subaddress string for the currently selected connection.

=CN_CHANNEL_TYPE
(channel type --)
sets the channel type in use for the current connection to #B_CHAN or #D_CHAN

?CN_CHANNEL_TYPE
(-- channel type)
returns the channel type in use for the current connection

=CN_CHANNEL_ NUM
(channel number --)
sets the B-channel number for the current connection. This is only meaningful when the channel type for the current connection is set to #B_CHAN.

?CN_CHANNEL_ NUM
(-- channel number)
returns the B-channel number for the current connection.

=CN_PD
(protocol discriminator --)
sets the protocol discriminator being used for multiplexing of the current connection.

?CN_PD
(-- protocol discriminator)
returns the protocol discriminator being used for multiplexing of the current connection.

=CN_CEI
(SAPI / CES --)
sets the Connection Endpoint Suffix (CES) and the Service Access Point Identifier (SAPI) for the currently selected connection.

?CN _CEI
(-- SAPI / CES)
return the Connection Endpoint Suffix (CES) and the Service Access Point Identifier (SAPI) for the currently selected connection.

6.5.3. Call Reference Management

=CR_LENGTH
(call ref length --)
sets the call reference length for use with all connections.

?CR_LENGTH
(-- call ref length)
returns the call reference length being used for all connections.

=CN_CR_SOURCE
(call ref source --)
sets the call reference source for the currently selected connection. The source may be specified as either #INCOMING or #OUTGOING , depending on whether the DUT or the K1195 initiated the call.

?CN_CR_SOURCE
(-- call ref source)
returns the call reference source for the currently selected connection.

=CN_CR_VALUE
(call ref value --)
sets the call reference value for the currently selected connection.

?CN_CR_VALUE
(-- call ref value)
returns the call reference value for the currently selected connection.

CR_VAL_MAX
(-- call ref value)
returns the maximum call reference value based on the current call reference length.

SELECT_CR
(-- call ref value)
Returns a call reference value which is not in use for any current outgoing calls, or 0 if none are available.

CN_INIT
(direction / SAPI --)
sets the SAPI, the call reference source, the call reference value, the protocol discriminators and the CES for the current connection. The direction may be specified as one of the constants #INCOMING or #OUTGOING. If the connection is incoming, the call reference is taken from the last decoded message. In addition, the current international and national protocol discriminators, the default IE fields, and the default IE strings for the current protocol variant are set for the current connection. Otherwise it is generated automatically by SELECT_CR. (See also section 6.3.6.6 "Connection Multiplexer")

CLEAR_CR
(--)
sets the call reference value for the current connection to #CREF_UNUSED which indicates that the call reference has been released but the connection is still active. This is done when a connection has been suspended.

6.5.4. Information Element Buffer and Parameter Management

?CN_IE_LIST **(-- ptr)**
returns the base address of the structure of IE parameters for the currently selected connection.

COPY_PARAMS **(srce CN / dest CN --)**
copies the entire set of IE octet field values from one connection data structure to another. #DECODER may be used as the source CN, indicating the string is to be copied from the decoder instead of another CN.

COPY_IE **(srce CN / dest CN / IE index --)**
copies only the octet field values for the specified IE from one connection data structure to another. #DECODER may be used as the source CN, indicating the string is to be copied from the decoder instead of another CN.

INIT_IE_BUFFERS **(--)**
clears all the IE buffer contents for the current connection.

LONG_BUFFER **(IE index --)**
assigns one of the three extended buffers (122 bytes) to the specified IE.

RESET_IE_MAP **(--)**
reassigns short buffers to all IEs for the current connection.

RESET_ALL_IE_MAPS **(--)**
reassigns short buffers to all IEs for all connections. This command should be used with caution.

?CN_IE_BUFFER **(IE index -- ptr or 0)**
returns the address of the buffer (used to store a canned IE) for the specified IE for the currently selected connection or 0 if the index was out of range.

COPY_BUFFERS **(srce CN / dest CN / --)**
copies all the IE buffer contents from one connection data structure to another. #DECODER may NOT be used here as the source CN.

COPY_IE_BUF **(srce CN / dest CN / IE index --)**
copies the specified IE buffer contents from one connection data structure to another. #DECODER may be used as the source CN, indicating the string is to be copied from the decoder instead of another CN.

6.5.5. B-Channel Management

BC_INIT (--)
used to initialise the B-channel table, such that all B-channels for these entries are flagged as unallocated and disconnected.

BC_ALLOC (bchan --)
flags the specified B-channel as being allocated, after performing range checking.

BC_DISC (bchan --)
flags the specified B-channel as being disconnected, after performing range checking.

BC_FREE (bchan --)
flags the specified B-channel as being deallocated, after performing range checking. Also flags it as being disconnected, if this has not already been done.

BC_CONN (bchan --)
flags the specified B-channel as being connected, after performing range checking. Also flags it as being allocated, if this has not already been done.

?BC_CONN (bchan -- flag)
returns 1 if the specified B-channel is flagged as being connected, 0 if flagged as being disconnected, and -1 if the B-channel is out of range.

?BC_ALLOC (bchan -- flag)
returns 1 if the specified B-channel is flagged as being allocated, 0 if flagged as being deallocated, and -1 if the B-channel is out of range.

?BC_SELECT (-- bchan)
returns the number of the first unallocated B-channel. If none of those available are unallocated, zero is returned.

SET_INTERFACE (interface --)
used to initialize the number of B-channels, the call reference length, and the connection and B-channel tables. The interface may be specified by the constants #PRIMARY (call reference of two octets, 30 B-channels) or #BASIC (call reference of one octet, 2 B-channels).

6.5.6. Miscellaneous

CR_TO_CN **(protocol discriminator / cref value / cref flag / SAPI / CES -- CN or -1)**
returns the connection number of the CN that the specified data pertains to. This CN is made the current connection. A -1 is returned if the data was not associated with any CN.

COPY_CN **(CN srce / CN dest --)**
copies all the connection data from one CN to another.

L3_MUX_ON **(--)**
switches on the Layer 3 multiplexer, allowing the Test Manager to handle up to eight simultaneous connections. When an event occurs, the appropriate connection is activated to handle it.

L3_MUX_OFF **(--)**
switches off the Layer 3 multiplexer, returning the Test Manager to handling a single connection.

SVC_ON **(--)**
switches on the supervisor control of keyboard events (function keys and string input) within a testscript, i.e. all keyboard events will be automatically handled by TM0.

SVC_OFF **(--)**
switches off the supervisor control of keyboard events (function keys and string input) within a testscript, i.e. all keyboard events will be handled by the currently active testmanager (default).

6.6. MESSAGE BUFFER POOL USER INTERFACE

The Message Pool Management is used to manage a collection of Layer 3 messages for simulation purposes. In the module, an array of Pool Entries is declared, each of which contains a name field (used to identify the message), a comment field, and a data field which contains the message itself. Both the name fields and the comment fields are stored in the Pool Entry as normal FORTH strings, while the data field uses the first two bytes to denote the message length. The maximum number of buffers in the Message Buffer Pool is specified by the constant N#MSG_POOL_ENTRIES which is currently set to 60.

The length of a message which may be stored in a single buffer is 130 octets. However, an extension mechanism exists whereby two adjacent buffers may be used to store an extended message (up to 262 bytes). The operation of this mechanism is fully automatic, and will occur whenever a message is stored which is too large to fit in a single buffer. An empty buffer, or a message contained in one or two buffers is called a Pool Entry.

The user interface to the Message Buffer Pool Management enables the user to access specific, predefined Layer 3 messages in Test Manager scenarios. These predefined messages are made with the Coder, using either the Menu Building System (see section 3) or the Programming Method (see section 6.7) and are stored to disk using the Message Buffer Pool Management.

Specifically the Message Buffer Pool provides FORTH commands to address the name, comment, and data fields of any Entry in the message pool. It is also possible to initialize single Pool Entries or the entire pool. Individual message strings can be loaded or recalled to the pool while the entire message pool can be stored or recalled from disk files. In addition, commands exist which permit the selection of a message from the pool by specifying the name field.

The following commands provide the User Interface to the Message Pool. (The information in parenthesis indicates the before and after condition of the FORTH stack.)

?MESSAGE **(name string address -- addr or 0)**
Returns the data field address of the first message pool entry whose name field matches that specific or 0 if none matched. An error message is generated if no match is found.

?MESSAGE_NUMBER **(name string address -- n)**
Returns the number of the first message pool entry whose name field matches that specified or -1 if none matched. An error message is generated if no match is found.

?MESSAGE_NAME **(n -- address of nth name field or 0)**
Returns the address of the name field of the nth message pool entry or returns zero and displays an error message if n is invalid.

?MESSAGE_COMMENT **(n -- address of nth comment field or 0)**
Returns the address of the comment field of the nth message pool entry or returns zero and displays an error message if n is invalid.

?MESSAGE_DATA **(n -- address of nth data field or 0)**
Returns the address of the data field of the nth message pool entry or returns zero and displays an error message if n is invalid.

?MESSAGE_FREE **(n -- flag)**
Returns true if the nth message pool entry is currently not in use, either for a complete message or for part of an extended message, or false if n is invalid.

STORE_MESSAGE **(addr / length / n --)**
Copies a message starting at 'addr' and of 'length' bytes into the data field of the nth message pool entry. If the specified message is too large to fit in one buffer entry, this command automatically extends the buffer to use the next entry. If the next entry is unavailable, an error message is given and no part of the message is stored.

RECALL_MESSAGE **(addr / n -- length or -1)**
Copies the string located in the data field of the nth message pool entry to 'addr'. The length in bytes of the message is left on the stack, or -1 in case of failure, i.e. if n was an invalid buffer index.

CLEAR_MESSAGE **(n --)**
Fills the name, comment, and data fields of the nth message pool entry with nulls. If the message was extended, both the nth and n+1th entries are filled with nulls.

SAVE_MESSAGES **(filename string address --)**
Saves the entire message buffer pool to the indicated file. Error messages are given in case of failure.

LOAD_MESSAGES **(filename string address --)**
Loads the message buffer pool from the indicated file after checking the file size and type. Error messages are given in case of failure.

CLEAR_POOL **(--)**
Fills the name,comment, and data fields of all message pool entries with nulls. This command should be used with caution.

ALLOC_MESSAGE **(-- n)**
Returns the number of the first pool entry not currently in use, with the next entry also unused, in case it needs to be used for an extended message. A -1 is returned if no such entry can be found in the pool.

COMPACT_POOL **(-- n)**
Shuffles all pool entries which are in use to the bottom of the pool i.e. closer to entry 0. The number of pool entries in use is returned.

CONCAT_POOLS **(filename string address --)**
Compacts the existing message pool, and concatenates the pool contained in the specified file to the existing one. If the two pools are too large to fit together in the one pool, the file pool is truncated to fit, and an error message is given.

INSERT_ENTRY **(n --)**
Inserts a blank pool entry before the nth entry by shuffling the nth and subsequent entries upwards in the pool. If the last entry in the pool is in use, or if n is an invalid index, the pool remains unaltered and an error message is given.

COPY_MESSAGE **(name string / n --)**
Copy the entire contents of the pool entry specified by the name string into the nth pool entry. An error message is given if unsuccessful.

MOVE_MESSAGE **(name string / n --)**
Copy the entire contents of the pool entry specified by the name string into the nth pool entry, and delete the source entry. An error message is given if unsuccessful.

=POOL_CR_VALUE **(message name / call ref value --)**
Set the call reference value for the pool message of the given name. If the dummy call reference is in use for the specified message, the entry remains unchanged and an error message is generated. An error message is also generated if no message of the given name exists in the pool.

?POOL_CR_VALUE **(message name -- call ref value or 0)**
Return the call reference value used by the pool message of the given name. 0 is returned and an error message is generated if the message is not found or the dummy call reference is in use for the message.

=POOL_CR_FLAG **(message name / call ref flag --)**
Set the call reference flag for the pool message of the given name. If the dummy call reference is in use for the specified message, the entry remains unchanged and an error message is generated. An error message is also generated if no message of the given name exists in the pool.

?POOL_CR_FLAG **(message name -- call ref flag or -1)**
Return the call reference flag used by the pool message of the given name. -1 is returned and an error message is generated if the message is not found or the dummy call reference is in use for the message.

ALTER_CALL_REF **(name string / call ref. flag / call ref. value --)**
Changes the call reference value and flag of the pool message with the specified name to those supplied on the stack. The rest of the message is unaltered. An error message is given if the message is not found or the dummy call reference is in use for the message.

The following five commands are only available in auto mode.

EDIT_MSG **(name string -- flag)**
Takes the specified message from the pool and puts it into the coder area to allow to modify the message. A true flag is returned if no error has occurred.
Note: The message will be completely decoded, i.e. all decoder variables (e.g. $MSG_ID) will be overwritten by the message.

STORE_MSG **(name string --)**
Writes the coder area into the pool entry specified by the name string. If the message did not exist before, the entry will be created.

GET_RX_MSG **(---)**
Takes the last received message and puts it into the coder area to allow to modify the message.

APPEND_IE **(IE-identifier --)**
Includes specified IE into the message in the coder area.

DELETE_IE **(IE-identifier --)**
Removes specified IE from the message in the coder area.

Example :

```
" SETUP" EDIT_MSG                                ( take " SETUP" from pool )
IF
          I#CALLING_NUM ?L3_IECOUNT 0=           ( insert I#CALLING_NUM if not )
          IF                                     ( already included )
                    " 12345" *COD ->CLGN_NUMBER !STRING
                    I#CALLING_NUM APPEND_IE
          ENDIF

          I#DISPLAY ?L3_IE_COUNT                 ( delete I#DISPLAY if included)
          IF
                    I#DISPLAY DELETE_IE
          ENDIF

          I#BEARER_CAP ?L3_IE_COUNT              ( modify I#BEARER_CAP if )
          IF                                     ( included )
                    #CCITT    *COD ->BC_CODING_STANDARD !
                    #SPEECH   *COD ->BC_TRANSFER_CAP !
          ENDIF

          " SETUP" STORE_MSG                     ( store " SETUP" )
ENDIF
```

SEND_MESSAGE **(name string --)**
Sends the Layer 3 message in the pool with the specified name field using an I frame. An error message is given if the specified message cannot be found.

SEND_UNIT_MESSAGE **(name string --)**
Sends the Layer 3 message in the pool with the specified name field using an UI frame. An error message is given if the specified message cannot be found.

NOTE: The User Interface words "SEND_MESSAGE" and "SEND_UNIT_MESSAGE" are more complex than described here, for the complete description refer to "Sending Messages" Section 6.4.3.

6.7. PROGRAMMING METHOD OF GENERATING MESSAGES

This method of message generation allows the user to produce Layer 3 messages using FORTH commands to define the contents and structure of a message. Commands are provided to specify the message type to be built and the information elements to be included in that message. This allows the user to easily build both correct and incorrect messages for use with the K1197 Simulator software packages. (The instructions on how to modify the contents of the fields within the information elements is covered in the section on "Information Elements Generation Commands").

The Message Building System uses a set of variables and buffers for the current connection when building the messages. (For details regarding connections see the ISDN LAYER 3 SIMULATION OVERVIEW). The user generates messages by specifying the message type and the Information Elements required. The Protocol Discriminator, the call reference value and the call reference flag are all derived from the variables for the current Connection. The information elements which are included in the message are defined by the user or are the default power-up system values.

The programming method has an automatic mode in which a message will be generated with the IEs sorted into the correct order and SHIFT IEs inserted to handle any national IEs. It also has a manual mode in which a message is generated with the IEs included in the order specified by the user.

6.7.1. Message Generation Commands

Messages are generated using the word **MESSAGE>** and one of the words **<SEND**, **<SEND_UNIT** or **<STORE**. **<SEND** and **<SEND_UNIT** generate a message and send it in an I frame or in a UI frame. **<STORE** generates a message and stores it in a user specified Pool Buffer.

The word **MESSAGE>** is passed a message identifier that is used to determine which type of message to generate (see the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual for a list of message identifiers). The words **<SEND**, **<SEND_UNIT** and **<STORE** use a list of IE identifiers which define which IEs are to be included in the message. These identifiers are used to access the IE buffers where the IEs have previously been generated. The contents of the buffers are used to build up the message. The order in which the IEs are built into the message buffer is the same as defined in the list of IE indices, except when in the automatic mode. In this mode the IEs are sorted into their correct order.

The words **<SEND** and **<SEND_UNIT** generate the message using the list of IE identifiers as described above, and then send the message in an I frame, in the case of **<SEND**, and in a UI frame in the case of **<SEND_UNIT** on the currently active data link. (See LAYER 2 EMULATION for the commands **DL_DATA_REQ** and **DL_UNIT_DATA_REQ**).

NOTE: The commands "<SEND" and "<SEND_UNIT" are more complex than described here, for a complete description see "Sending Messages" Section 6.4.3.

The word <STORE generates the message in the same manner as <SEND and <SEND_UNIT, but differs from these in that it must be supplied a buffer pool name. (The buffer pool is a collection of 60 buffers which can be used to for the storage of generated messages). It will then check the Message Buffer Pool for a buffer with the name described by the supplied string and will store the generated message in this buffer. If no buffer with this name exists, then a new buffer will be used and will be given the name defined by the string. The built message is then stored in this new buffer.

Example:

```
M#SETUP_ACK  MESSAGE>
             I#CHANNEL_ID        ( include Channel Id IE : 1st IE )
             I#PROGRESS_IND      ( include Progress Indicator IE: 2nd IE )
             I#DISPLAY           ( include Display IE : 3rd IE )
             I#SIGNAL            ( include Signal IE  : 4th IE )
             I#TEMP0             ( include IE in temporary buffer 0 )
" SETUP_ACK" <STORE              ( store message in specified buffer pool )
```

Note: A message which should contain IEs of temporary buffer (e.g. I#TEMP0) can only be generated in manual mode. In automatic mode, these IEs will be ignored.

6.7.2. Automatic Message Generation Mode

The automatic mode of message generation is entered by using the command AUTO_MODE and can be left by using the command MAN_MODE:

AUTO_MODE	(--)	Set the automaotic mode for message building.
MAN_MODE	(--)	Set the manual mode for message building.

IE generation and manipulation is the same in the automatic mode as in the manual mode. The same words are used for generating the messages and for sending or storing the generated message. The difference between the two modes is that the IE identifier list that is supplied to <SEND, <SEND_UNIT and <STORE is not order dependent in the automatic mode, and that the user need not include any SHIFT IEs. The words <SEND, <SEND_UNIT and <STORE will generate the message in automatic mode by including the IEs in the correct order and including any SHIFT IEs that may be needed.

Note that the automatic mode is limited to only being able to include one IE of each type, and cannot access the temporary IE buffers.

Example:

```
AUTO_MODE                        ( enter automatic mode )
M#ALERT   MESSAGE>               ( generate ALERT message )
          I#UU_INFO              ( include User User Info IE )
          I#CHANNEL_ID           ( include Channel Id IE )
          I#PROGRESS_IND         ( include Progress Indication IE )
          I#TERM_CAPAB           ( include Terminal Capabilities IE )
<SEND              ( sort IEs, generate message and send in an I frame )
MAN_MODE
```

6.7.3. Information Element Generation Commands

Information elements are generated in a similar way to which messages are generated in the programming method. Commands are provided to indicate the IE names, the octets to be included and the contents of each field (using meaningfully named constants).

Information elements are generated by the user from the data which is stored in variables within the K1197 Simulator/Monitor software. These variables contain the data for all information element fields as well as information about which of the various octets of each particular information element are to be included. Both the data for inclusion in the fields and the octet structure information can be altered by the user allowing the user to generate messages with any structure and data content. Note that the K1197 Simulator/Monitor software has default settings for all these variables.

The words **ELEMENT>** and **<ELEMENT** are used to generate an information element and are used when modifying the structure and data of an IE. The word **ELEMENT>** is used to start IE generation and must be passed an IE identifier. (See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual for a list of IE identifiers). The **<ELEMENT** command uses the data supplied by the user (or any previous data settings), after the Programming Information Element Generation **ELEMENT>** command to generate the specified IE. The generated IE will be stored in a buffer, so it can be used later when generating messages. Each IE type (eg. Bearer Capability) has a buffer assigned to it for the storage of generated IEs. There are an additional 8 temporary buffers for storing temporary IEs. These temporary buffers can be used for generating messages that the user requires to contain multiple versions of the same IE type. The extra copies are stored in the temporary buffers, and these are then included in the IE identifier list when the message is generated.

The user can modify the structure of an IE using the following commands after the command **ELEMENT>**:

ALL_INCLUDED **(--)**
will change the IE structure so that all possible octets will be included when the IE is generated.

ALL_EXCLUDED **(--)**
will cause none of the octets in the IE to be included. Only the IE identifier and an IE length = 0 will be included when the IE is generated.

INCLUDED **(Octet-identifier --)**
will change the IE structure so that the specified octet will be included when the IE is generated.

EXCLUDED **(Octet-identifier --)**
will cause the specified octet to be excluded when the IE is generated.

If none of these words are executed, then the structure of the IE will not be changed from the last time it was set. The extension bits (where used) within the IE will be automatically generated when the IE is generated. The field data can be modified and examined using the field selectors and the constant ***COD**. This constant points to the variables for the current connection (see section 6.3.6). The FORTH commands ! and **!STRING** can be used to store numeric values and strings into the various IE Field data variables. (Note: *COD is the Message Building System equivalent of *DEC. See the ISDN LAYER 3 (INTER)NATIONAL VARIANT Manual and the ISDN TEST MANAGER Manual for the definition of field selectors and *DEC).

LOAD_IE_BUFFER **(temp. buffer id --)**

The buffer that will contain the generated IE is selected by ELEMENT> using the user supplied IE identifier to identify the IE specific buffer. If the user wishes to generate an IE into one of the temporary buffers, the command **LOAD_IE_BUFFER** must be executed before **<ELEMENT**, preceeded by the temporary buffer identifier. The temporary IE buffers can be specified by using one of the following IE buffer identifiers: **I#TEMP0, I#TEMP1, I#TEMP2, I#TEMP3, I#TEMP4, I#TEMP5, I#TEMP6** or **I#TEMP7**.

Note: A message which should contain IEs of temporary buffer can only be generated in manual mode. In automatic mode, these IEs will be ignored.

Example 1:

```
I#BEARER_CAP ELEMENT>                    ( build Bearer Cap with octets 3,4, )
                                         ( 4a & 4b specify IE structure )

        ALL_INCLUDED                     ( include all possible octets )
        OCTET_7 EXCLUDED                 ( exclude octet 7 )
        OCTET_6 EXCLUDED                 ( exclude octet 6 )
        OCTET_5A EXCLUDED                ( exclude octet 5A )
        OCTET_5 EXCLUDED                 ( exclude octet 5 )
                                         ( modify IE field data )

        #CCITT              *COD ->BC_CODING_STANDARD !          ( Octet 3 )
        #SPEECH             *COD ->BC_TRANSFER_CAP !

        #CIRCUIT_MODE       *COD ->BC_TRANSFER_MODE !            ( Octet 4 )
        #64KBIT/S *COD ->BC_TRANSFER_RATE !

        #DEFAULT  *COD ->BC_STRUCTURE !                          ( Octet 4a )
        #POINT_TO_POINT     *COD ->BC_CONFIGURATION !
        #DEMAND             *COD ->BC_ESTABLISHMENT !

        #BIDIRECT_SYMMETRIC  *COD ->BC_SYMMETRY !                ( Octet 4b )
        #CHANNEL_SIZE       *COD ->BC_TRANSFER_RATE_4B !

<ELEMENT                                 ( generate IE into Bearer Cap )
                                         ( default IE buffer )
```

Example 2:

```
I#CHANNEL_ID ELEMENT>                    ( build Channel Id IE with only new )
                                         ( channel map use previously specified )
                                         ( IE structure )

        *DEC ->CID_MAP @   ( fetch channel map from received message )
        *COD ->CID_MAP !   ( store in coder variable )

<ELEMENT                                 ( generate IE into Channel Id buffer )
```

Example 3:

```
I#DISPLAY ELEMENT>                    ( generate IE into a temporary buffer )
                                      ( use previously specified IE structure)

          " Test string for Display IE"        ( enter string to be stored )
          *COD_->D_DISPLAY !STRING             ( modify Display information )

          I#TEMPO LOAD_IE_BUFFER      ( specify temporary buffer 0 for IE
                                      ( generation )
<ELEMENT                              ( generate Display IE into temporary
                                      ( buffer 0 )
```

A. APPENDIX

A.1. STATES FOR THE LAYER 2, THE ASP AND THE ML

K1197	CCITT	Layer 2 STATE DEFINITION
1	1	TEI UNASSIGNED
2	2	ASSIGN AWAITING TEI
3	3	ESTABLISH AWAITING TEI
4	4	TEI ASSIGNED
50	5.0	AWAITING ESTABLISHMENT, Establish
51	5.1	AWAITING ESTABLISHMENT, Re-establish
52	5.2	AWAITING ESTABLISHMENT, Pending release
6	6	AWAITING RELEASE
		MULTIPLE FRAME ESTABLISHED
70	7.0	NORMAL, Normal
71	7.1	NORMAL, Reject recovery
72	7.2	NORMAL, Own receiver busy
73	7.3	NORMAL, Reject recovery and own receiver busy
74	7.4	PEER RECEIVER BUSY, normal
75	7.5	PEER RECEIVER BUSY, Reject recovery
76	7.6	PEER RECEIVER BUSY, Own receiver busy
77	7.7	PEER RECEIVER BUSY, Reject revovery and own receiver busy
		TIMER RECOVERY
80	8.0	NORMAL, Normal
81	8.1	NORMAL, Reject recovery
82	8.2	NORMAL, Own receiver busy
83	8.3	NORMAL, Reject recovery and own receiver busy
84	8.4	PEER RECEIVER BUSY, normal
85	8.5	PEER RECEIVER BUSY, Reject recovery
86	8.6	PEER RECEIVER BUSY, Own receiver busy
87	8.7	PEER RECEIVER BUSY, Reject revovery and own receiver busy

STATE	ASP STATE DEFINITION
1	IDLE
2	TEI BEING ASSIGNED
3	TEI CHECKING

STATE	ML STATE DEFINITION
1	TEI UNASSIGNED
2	TEI BEING RESPONSED
3	XID RESPONSE WAITING
4	TEI ASSIGNED
5	IDENTITY VERIFY

A.2. PRIMITIVE AND CONSTANT NAMES

	REQUEST	INDICATION	RESPONSE	CONFIRM
L3--L2	(L3 TO L2)	(L2 TO L3)	(L3 TO L2)	(L2 TO L3)
DL_ESTABLISH	DL-EST#REQ	DL-EST#IND	---	DL-EST#CONF
DL_RELEASE	DL-REL#REQ	DL-REL#IND	---	DL-REL#CONF
DL_DATA	DL-DATA#REQ	DL-DATA#IND	---	DL-DATA#CONF
DL_UNIT_DATA	DL-UDATA#REQ	DL-UDATA#IND	---	---
DL_SET_BUSY	DL-SETBUSY#REQ	---	---	---
DL_CLR_BUSY	DL-CLRBUSY#REQ	---	---	---
ML--L2	(ML TO L2)	(L2 TO ML)	(ML TO L2)	(L2 TO ML)
'MDL_ASSIGN'	MDL-ASSIGN#REQ	MDL-ASSIGN#IND	---	---
MDL_REMOVE	MDL-REM#REQ	---	---	---
MDL_ERROR	---	MDL-ERR#IND	MDL-ERR#RESP	---
MDL_UNIT_DATA	MDL-UDATA#REQ	MDL-UDATA#IND	---	---
'MDL_XID'	MDL-XID-REQ#	MDL-XID-IND#	MDL-XID-RESP#	MDL-XID-CONF#
L2--L1	(L2 TO L1)	(L1 TO L2)	(L2 TO L1)	(L1 TO L2)
'PH_DATA'	PH-DATA#REQ	PH-DATA#IND	---	---
PH_ACTIVATE	PH-ACT#REQ	PH-ACT#IND	---	---
PH_DEACTIVATE	PH-DEACT#REQ	PH-DEACT#IND	---	---
ML--L1	(ML TO L1)	(L1 TO ML)	(ML TO L1)	(L1 TO ML)
'MPH_ERROR'	---	MPH-ERR#IND	MPH-ERR#RESP	---

Note: The primitives enclosed in " ' " are not implemented in the L2 Reference Emulation.

INDEX

Symbols

?B/D_CALLED, 6-19
?B/D_CALLING, 6-19
?B/D_CONN_CNF, 6-18
?B/D_CONN_IND, 6-18
?B/D_DET_IND, 6-18
?B/D_DISC_CNF, 6-18
?B/D_DISC_IND, 6-18
?B/D_UPTR, 6-19
?BC_ALLOC, 6-12, 6-26
?BC_CONN, 6-12, 6-26
?BC_SELECT, 6-12, 6-26
?CN, 6-8, 6-22
?CN_CALLED_NUM, 6-9, 6-23
?CN_CALLED_SAD, 6-9, 6-23
?CN_CALLING_NUM, 6-9, 6-23
?CN_CALLING_SAD, 6-9, 6-23
?CN_CEI, 6-9, 6-24
?CN_CHANNEL_NUM, 6-9, 6-23
?CN_CHANNEL_TYPE, 6-9, 6-23
?CN_CR_SOURCE, 6-10, 6-24
?CN_CR_VALUE, 6-24
?CN_IE_BUFFER, 6-11, 6-25
?CN_IE_LIST, 6-25
?CN_PD, 6-9, 6-23
?CN_TIMER, 6-13, 6-20
?CN_TIMER_DUR, 6-13
?CR_LENGTH, 6-10, 6-24
?L2_ON, 4-14
?L3_ID_COUNT, 6-20
?L3_IE, 6-20
?L3_IE_ORDER, 6-20
?L3_MAND_IE, 6-20
?L3_MATCH, 6-20
?L3_MSG, 6-20
?L3_OCTET, 6-20
?L3_OPT_IE, 6-20
?L3_UNEXP_IE, 6-20
?L3_VALID_MSG, 6-20
?MESSAGE, 6-28
?MESSAGE_COMMENT, 6-28
?MESSAGE_DATA, 6-28
?MESSAGE_FREE, 6-29
?MESSAGE_NAME, 6-28
?MESSAGE_NUMBER, 6-28
?ML_ON, 4-15
?POOL_CR_FLAG, 6-30
?POOL_CR_VALUE, 6-30
?PTP, 4-14
/CN_CR_VALUE, 6-10
=B/D_CALLED, 6-19
=B/D_CALLING, 6-19
=B/D_UPTR, 6-19
=CN, 6-8, 6-22
=CN_CALLED_NUM, 6-9, 6-23
=CN_CALLED_SAD, 6-9, 6-23
=CN_CALLING_NUM, 6-9, 6-23
=CN_CALLING_SAD, 6-9, 6-23
=CN_CEI, 6-9, 6-23
=CN_CHANNEL_NUM, 6-9, 6-23
=CN_CHANNEL_TYPE, 6-9, 6-23
=CN_CR_SOURCE, 6-10, 6-24
=CN_CR_VALUE, 6-10, 6-24
=CN_PD, 6-9, 6-23
=CN_TIMER_DUR, 6-13
=CR_LENGTH, 6-10, 6-24
=ICHAR, 4-16
=NAT_BITS, 4-16
=POOL_CR_FLAG, 6-30
=POOL_CR_VALUE, 6-30

A

AI[], 4-22
ALL_EXCLUDED, 6-35
ALL_INCLUDED, 6-35
ALLOC_MESSAGE, 6-29
ALTER_CALL_REF, 6-30
APPEND_IE, 6-31
ASPST, 4-8
ATEI, 4-12
AUTO_BROADCAST, 6-21
AUTO_HDR_UPDATE, 6-21
AUTO_MODE, 6-34

AUTO_UI_SETUP, 6-21

B

B/D_CONN_REQ, 6-18
B/D_CONN_RES, 6-18
B/D_DISC_REQ, 6-18
B/D_DISC_RES, 6-18
BALM_OFF, 4-16
BALM_ON, 4-16
BC_ALLOC, 6-12, 6-26
BC_CONN, 6-12, 6-26
BC_DISC, 6-12, 6-26
BC_FREE, 6-12, 6-26
BC_INIT, 6-12, 6-26
BER, 4-12
BLOCK-COUNT, 4-20
BYTES, 4-10

C

CES, 4-12
CES=0, 4-12
CES=1..7, 4-12
CLEAR_CR, 6-10, 6-24
CLEAR_MESSAGE, 6-29
CLEAR_POOL, 6-29
CLR POOL, 16
CLREC, 4-12
CMD/RESP*, 4-21
CN_INIT, 6-10, 6-15, 6-24
CN_START_TIMER, 6-13
CN_STOP_TIMER, 6-13
CN0, 6-8
CN0 ... CN7, 6-22
CN1, 6-8
CN2, 6-8
CN3, 6-8
CN4, 6-8
CN5, 6-8
CN6, 6-8
CN7, 6-8
COMPACT_POOL, 6-29
CONCAT, 16
CONCAT_POOLS, 6-29
COPY_BUFFERS, 6-11, 6-25
COPY_CN, 6-27
COPY_IE, 6-11, 6-25
COPY_IE_BUF, 6-11, 6-25
COPY_MESSAGE, 6-30
COPY_PARAMS, 6-11, 6-25
CR_TO_CN, 6-27
CR_VAL_MAX, 6-10, 6-24
CRC_ERROR, 4-11

D

DATA-STATUS, 4-23
DELETE_IE, 6-31
DISC, 4-1
DL_CLR_BUSY, 4-5
DL_DATA, 4-5
DL_DATA_REQ, 4-5, 6-4
DL_ESTABLISH, 4-5, 6-4
DL_RELEASE, 4-5, 6-4
DL_SET_BUSY, 4-5
DL_UNIT_DATA, 4-5
DL_UNIT_DATA_REQ, 4-5, 6-4
DM, 4-1
DO_ABORT, 4-11

E

EDIT_MSG, 6-31
END-TIME, 4-20
EVENT-TYPE, 4-20
EXCLUDED, 6-35
EXTERN_CLK, 4-17, 4-18

F

F0, 4-9
F1, 4-9
FER, 4-12
FERR=EVEN, 4-16
FERR=ODD, 4-16
FRAME-ID, 4-22

FRMR-C/R-BIT*, 4-21
FRMR-CNTL1*, 4-21
FRMR-CNTL2*, 4-21
FRMR-VR*, 4-21
FRMR-VS*, 4-21
FRMRW*, 4-21
FRMRX*, 4-21
FRMRY*, 4-21
FRMRZ*, 4-21

G

GET_RX_MSG, 6-31
GOOD_CRC, 4-11

I

IDENTITY_ASSIGNED, 4-2
IDENTITY_CHECK_REQ, 4-2
IDENTITY_CHECK_RESP, 4-2
IDENTITY_DENIED, 4-2
IDENTITY_REMOVE, 4-2
IDENTITY_REQUEST, 4-2
IDENTITY_VERIFY, 4-2
IMPED=100, 4-17
IMPED=HIGH, 4-17
INCLUDED, 6-35
INIT_DEFAULT_TIMERS, 6-13
INIT_IE_BUFFERS, 6-11, 6-25
INSERT_ENTRY, 6-30
INT_BIT=0, 4-16
INT_BIT=1, 4-16
INTERN_CLK, 4-17

K

K, 4-8
K*, 4-19

L

L2-LENGTH, 4-20
L2-POINTER, 4-20
L2_EXEC, 4-14
L2_OFF, 4-14
L2_ON, 4-14
L2_STATE_TRC, 4-15
L2_TIMER_TRC, 4-15
L2ST, 4-8
L3-LENGTH, 4-20
L3-POINTER, 4-20
L3_MUX_OFF, 6-14, 6-27
L3_MUX_ON, 6-14, 6-27
LOAD POOL, 16
LOAD_IE_BUFFER, 6-36
LOAD_MESSAGES, 6-29
LONG_BUFFER, 6-11, 6-25

M

MADDR1*, 4-20
MADDR2*, 4-20
MAKE_DATA1, 4-13
MAKE_DATA1 / ... / MAKE_DATA8, 4-10
MAKE_DATA2/.../MAKE_DATA8, 4-13
MAN_MODE, 6-34
MCONTROL1*, 4-21
MCONTROL2*, 4-21
MDL_ALLOCATE, 4-6, 6-4
MDL_ASSIGN_I, 4-6
MDL_ASSIGN_R, 4-6
MDL_CURRENT, 4-6, 6-4
MDL_DEALLOCATE, 4-6, 6-4
MDL_ERROR_TRC, 4-15
MDL_EXISTED, 4-6, 6-4
MDL_INUSE, 6-4
MDL_REMOVE, 4-6, 6-4
MDL_SELECT, 4-6, 6-4
MDL_UNIT_DATA, 4-5
ML_EXEC, 4-15
ML_OFF, 4-15
ML_ON, 4-14
MLST, 4-8
MNR*, 4-21
MNS*, 4-21
MOVE_MESSAGE, 6-30
MP/F-BIT*, 4-21
MSAPI*, 4-21

MTEI*, 4-21

N

N200, 4-9
N200*, 4-19
N201, 4-9
N201-TX*, 4-19
N201*, 4-19
N201TX, 4-9
N202, 4-9
N202*, 4-19
NEW_SPEED, 4-18
NEW_TEI, 4-12
NM20, 4-9
NR, 4-7
NS, 4-7
NTWK_EMUL, 4-14

P

P*SENT, 4-19
P0, 4-9
P1, 4-9
PH_ACTIVATE, 4-6
PH_DEACTIVATE, 4-6
PIGGY*, 4-21
PKT, 4-10
PKTS, 4-10
PORT-ID, 4-20
POWER_FEED_INVERS, 4-17
POWER_FEED_OFF, 4-17
POWER_FEED_ON, 4-17
PT-K, 4-8
PT-N201, 4-9
PT-N201TX, 4-9
PT-T200, 4-8
PT-T203, 4-8
PTMP, 4-14
PTP, 4-14

Q

QUIT_HEX, 16

R

RC, 4-7
RC-ML1*, 4-19
RC*, 4-19
RC20, 4-7
RC20*, 4-19
RC201, 4-7
RC201*, 4-19
RECALL_MESSAGE, 6-29
RESER_ALL_IE_MAPS, 6-25
RESET_ALL_IE_MAPS, 6-11
RESET_IE_MAP, 6-11, 6-25

S

S-DATA-AUTO*, 4-13
S_BUS_EXT, 4-17
S_BUS_PTP, 4-17
S_BUS_SHORT, 4-17
S_DISC, 4-3
S_DM, 4-3
S_FRMR, 4-3
S_I, 4-3
S_ID_ASSIGNED, 4-4
S_ID_CHECK_REQ, 4-4
S_ID_CHECK_RESP, 4-4
S_ID_DENIED, 4-4
S_ID_REMOVE, 4-4
S_ID_REQUEST, 4-4
S_ID_VERIFY, 4-4
S_REJ, 4-3
S_REJC, 4-3
S_RNR, 4-3
S_RNRC, 4-3
S_RR, 4-3
S_RRC, 4-3
S_SABME, 4-3
S_UA, 4-3
S_UI, 4-3

S_XID, 4-4
S_XID0, 4-4
S_XIDC, 4-4
S_XIDC0, 4-4
SABME, 4-1
SAPI, 4-12
SAPI*, 4-19
SAVE POOL, 16
SAVE_MESSAGES, 6-29
SELECT_CR, 6-10, 6-24
SEND_FRMR, 4-4
SEND_MESSAGE, 6-32
SEND_UNIT_MESSAGE, 6-32
SENDF, 4-3
SENDP, 4-1
SET_INTERFACE, 6-12, 6-26
SI1_BIT=0, 4-16
SI1_BIT=1, 4-16
SI2_BIT=0, 4-16
SI2_BIT=1, 4-16
SLAVE_CLK, 4-17
STANDARD_CLK, 4-18
START-TIME, 4-20
STATE-L2*, 4-19
STOP_SCANNING, 6-15
STORE_MESSAGE, 6-29
STORE_MSG, 6-31
SVC_OFF, 6-15, 6-27
SVC_ON, 6-15, 6-27

T

T200, 4-8
T201, 4-8
T202, 4-8
T203, 4-8
TEI, 4-12
TM20, 4-8

U

U/N, 4-8
UI-TYPE*, 4-22
UI_3, 4-1
UI_M, 4-1
USER_EMUL, 4-14
USR*NET, 4-19

V

VA, 4-7
VR, 4-7
VR*, 4-19
VS, 4-7
VS*, 4-19
VTEI*, 4-19

W

W0, 4-10
W1, 4-10
WRITE, 16

X

X0, 4-10
X1, 4-10
XID-K*, 4-21
XID-RX-N201*, 4-21
XID-T200*, 4-21
XID-TX-N201*, 4-21
XID_OFF, 5-3
XID_ON, 5-3
XIDC, 4-2
XIDV, 4-12

Y

Y0, 4-10
Y1, 4-10
YALM_OFF, 4-16
YALM_ON, 4-16

Z

Z0, 4-10
Z1, 4-10

ISDN
TEST MANAGER

User Manual for
Protocol Tester K1197

SIEMENS AG

Software Version : V3.0
System Version : MF 2.2

Revision Date : January 1991

1. INTRODUCTION 1-1

2. GENERAL 2-1

2.1. FUNCTION KEYS 2-1

2.2. CONTROL COMMANDS 2-2

2.2.1. GENERAL 2-2

2.2.2. MULTIPLE TESTSCENARIO SUPPORT 2-3

3. STATE MACHINE OVERVIEW 3-1

4. EVENT RECOGNITION 4-1

4.1. ?WAKEUP 4-1

4.2. ?COMMAND 4-1

4.3. ?RESPONSE 4-1

4.4. ?RX_FRAME 4-2

4.5. ?RX_MATCH 4-3

4.6. ?RX_SEARCH 4-3

4.7. ?ABORT 4-4

4.8. ?CRC_ERROR 4-4

4.9. ?TIMER 4-4

4.10. ?L1_EVENT 4-5

4.11. ?L2_SERVICE 4-5

4.12. ?PIGGYBACKED 4-6

4.13. ?TM-IPC 4-7

4.14. ?KEY 4-7

4.15. FK_F1..FK_F6 4-8

4.16. ?KEYBOARD 4-8

4.17. OTHER_EVENT 4-9

4.18. SELF DEFINED EVENT RECOGNITION 4-9

4.19. SELF DEFINED EVENT GENERATION 4-10

5. ACTION WORDS 5-1

5.1. DISPLAY 5-1

5.1.1. COMMAND LINE DISPLAY 5-1

5.1.2. STATUS LINE DISPLAY 5-1

5.2. FUNCTION KEYS 5-2

5.3. LAPSE TIMERS 5-2
5.4. INTERVAL TIMERS 5-3
5.5. IPC MESSAGE SENDING 5-3
5.6. BUFFER CREATION 5-4

6. TRACE REPORT COMMANDS 6-1

7. TEST MANAGER ACTIVATION 7-1

8. LAYER 3 DECODER VARIABLES 8-1

9. LAYER 3 DECODER COMMANDS 9-1

10. ACCESSING INFORMATION ELEMENT FIELD VALUES 10-1

11. TEST SCRIPT EXAMPLES 11-1

11.1. TEST0.F 11-1
11.2. TEST1.F 11-3
11.3. TEST2.F 11-6
11.4. TEST3.F 11-8
11.5. TEST4.F 11-10
11.6. TEST5.F 11-13
11.7. TEST6.F 11-16
11.8. TEST7.F 11-19
11.9. TEST8.F 11-22

INDEX I-1

1. INTRODUCTION

The ISDN Test Manager is used to execute a user-defined test scenario which may test the protocol and/or service conformance of an IUT (Implementation Under Test), or simply store or display the desired information of the monitored data. Typical information that might be stored and/or displayed includes throughput performance data and the occurence of a particular frame sequence. In order to set up a test scenario, a set of FORTH words is provided to assemble the test as a finite state machine.

The Test Manager can be invoked both by the Monitor and the Emulator. When it is invoked by the Monitor, it serves as an analytical tool which allows the user to collect specific data using a specified format on the desired device. When it is invoked by the Emulator, it not only has all the analytical power, but also the capability to send tailored frames to the test suite so that both protocol and service conformance of the test suite can be examined.

The user can terminate the Test Manager at any time and resume the execution at any reasonable configuration. (For example, it is not reasonable to examine incoming frames when the monitor is in halt mode or to examine playback data when the monitor is reading data from buffer or disk. Such combinations are ignored by the Test Manager.)

The current implementation provides 8 "virtual" Test Managers, i.e. up to 8 dedicated Test Scenarios can be loaded. The user can select a specific Test Scenario by switching to the corresponding Test Manager. Virtual Test Managers can communicate with each other by sending self-defined messages via an IPC mechanism.

The ISDN Software provides a module called "Layer 3 Message Decoder", which enables the user to detect specific Layer 3 signalling events within Test Manager scenarios.

The Layer 3 Message Decoder also provides a simple, structured interface for user access of decoded information within Test Manager scenarios. Protocol discriminator, call reference and message type information is available in normal FORTH variables. Section 8 describes these variables and their predefined values.

The structure of Layer 3 Signalling messages and information elements is very complex and detailed protocol knowledge is required in order to establish test scenarios. Messages for example can contain a variable number of information elements, some of which are mandatory and others which are optional or not allowed. The sequence of information elements for a particular codeset is predefined but may be interspersed with information elements from other codesets. Furthermore multiple Information Elements of the same type may occur in a message and may have a variable structure depending on inclusion or exclusion of particular octets. To overcome these problems and simplify the method by which this information is obtained the Layer 3 Message Decoder provides customized functions to determine

- if all mandatory information elements are present within a message
- if any optional information elements are present within a message
- if any unexpected information elements are present within a message
- if all information elements occur in the correct order within the message
- how often a particular information element occurs within a message

- if it is possible to decode the nth occurrence of an information element
- if a particular octet was present in the information element
- the type of error occurred while decoding the information element

These functions are described in Section 9 of this document.

However information related to information elements requires use of a larger data structure, called the Information Element Parameter List, which provides field values for each octet contained in an Information Element. Access to these field values is by use of a predefined constant ***DEC** and a number of selectors which identify the addressed parameter. See Section 10 for a detailed explanation and examples.

Additionally the Layer 3 Message Decoder provides predefined constants which may be used to identify particular information element field values. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values within Test Manager scenarios. Therefore the test scenarios are easier to read and understand. This set of constants and words is different for each specific National Variant and is described in the ISDN NATIONAL VARIANTS Manual part.

Note: Examples in this document are related to a specific National variant, which can differ from your specific Software version.

2. GENERAL

The Test Manager scenarios are organized into state machines to allow for clearly defined test procedures. Each Test Manager state machine has its own current state. Events that are detected in the current state will cause execution of the code defined in the corresponding STATE{...}STATE part, that means, if the event matches one of the event recognition words programmed in the "STATE" section, the corresponding ACTION{...}ACTION code is executed and entry to a new state may be done there if desired. Entry to a new state in this context means loading a new current state, which is executed with the **next** event.

Note: Since there are up to 8 concurrent Test Managers, the above applies to the currently active Test Manager.

2.1. FUNCTION KEYS

The ISDN menus provide a capability for loading the Test Manager. A set of function keys is reserved for use with test manager scenarios. This function keys set is reached by pressing **f8** in the default function key line (labelled: "TM_KEYS"). They are initially displayed as follows:

To execute a test script file, use the editor to create a file containing the state definitions, then activate the menu system (**f1**), go to the TEST MANAGER menu, select the drive to load the file from, select the "virtual" Test Manager, enter the file name, move the scroll bar to the field "LOAD TESTMANAGER" and press return.

To invoke the loaded test script, press function key **f7** in the Test Manager function key set.

To stop the execution of a test script, press function key **f7** again (now labelled "TM(n)STOP", where n is the number of the currently active Test Manager). This will terminate the Test Manager.

The function keys as well as most of the following commands apply to the currently active Test Manager, i.e. the actions are performed for the active Test Manager. Switching between Test Managers can be done by using the commands described in section 2.2.2.

Note: Loading of test scenarios can be accelerated by switching on the *fast compile mode*. Type **ON QUICKY** to enable the fast compile mode, **OFF QUICKY** to switch it off. Before saving or loading a binary (EXECB, SAVEB) make sure, the fast compile mode is switched off.

2.2. CONTROL COMMANDS

2.2.1. GENERAL

Since the software supports multiple Test Managers (Testscenarios), the user must switch to the desired Test Manager prior loading/activating a scenario (see section 2.2.2). The default Test Manager on system startup is Test Manager 0. The following commands apply to the currently active Test Manager.

TM_RUN **(--)**
will start the state machine running at state 0. The "STATE_INIT" code for state 0 (if there is some) will be executed immediately. The events in state 0 will be checked and when any of them happen, the defined action will be executed.

TM_STOP **(--)**
will terminate the operation of the Test Manager state machine.

EXECTS **(filename address --)**
will load the specified disk file in the Test Manager dictionary. This file must be a compiled binary Test Manager dictionary which was saved with the **SAVETS** command. The previous Test Manager dictionary will be overwritten. The command also verifies that the operating system and application software presently in the K1197 are compatible with the one used to compile and save the dictionary. Usage: " Filename" EXECTS

SAVETS **(filename address --)**
will save the presently compiled Test Manager dictionary as a disk file using the specified filename. This file would be reloaded using **EXECTS.**
Usage: " Filename" SAVETS

EXECF **(fielname address --)**
will load and compile files. It will use **EXEC** to compile source files or **EXECTS** to load the Test Manager dictionary which had been saved with **SAVETS.**
Usage: " Filename" EXECF.

NEW_TM **(filename address --)**
will do an EXECF to load and compile the specified file, then will execute **TM_RUN** to start the Test Manager for the new scenario running at State 0. The NEW_TM command can be included as an action of one scenario to load and execute the next scenario. The calling Testscenario is immediately terminated when calling NEW_TM. Usage: "Filename" NEW_TM
Note: NEW_TM must not be called outside the Test Manager (i.e. other than outside an ACTION{ }ACTION)

TCLR **(--)**
will initialize the state machine data structure. Any existing Test Senarios for the currently active TM will be cleared. The current state is set to 0. It is advisable to start new test scenarios with a TCLR command before defining the new state machine.

TM.RAM (--)
will display on the command line the memory usage of the Test Manager.

WAKEUP_ON (--)
will set a status that indicates that a special Timer is to be started for 100 Milliseconds when the **TM_RUN** is executed. The **?WAKEUP** event then will happen in State 0 and will start the Scenario execution.

WAKEUP_OFF (--)
will deselect the wakeup timer for **TM_RUN.**

2.2.2. MULTIPLE TESTSCENARIO SUPPORT

The following commands describe how to switch between Test Managers. If you don't want to use multiple Test Manager you can ignore these commands.

=TM (n --)
Switchs to the Test Manager n. (n=0..7)

Note: No range check is done for =TM, ensure n=0..7 !

?TM (-- n)
Returns the number of the current Test Manager.

TM0 (--)
Switches to Test Manager 0.

TM1..7 (--)
Switches to Test Manager 1..7 respectively.

GLOB_TM_RUN (--)
Performs a **TM_RUN** for all Test Managers (see the TM_RUN command above).

GLOB_TM_STOP (--)
Performs a **TM_STOP** for all Test Managers (see the TM_STOP command above).

GLOB_TCLR (--)
Performs a **TCLR** for all Test Managers (see the TCLR command above). It is advisable to execute a GLOB_TCLR before loading the first scenario.

3. STATE MACHINE OVERVIEW

The following is a set of rules that apply to the creation of state machines within a Test Manager scenario. There can only be one state machine in existence at any one time.

The state machine is implemented using FORTH words which cause compilation to be re-directed to a pseudo dictionary with additional control code added to perform the state execution.

The pseudo dictionary is an array in the memory. Thus, there is a limit to the size of the total state machine. 256 states are presently supported for each Test Manager.

Up to 256 states can be defined with state numbers of 0 to 255. States do not have to be defined in any particular sequence, ie. state 0, then state 1, state 2, etc. States do not have to be defined at the time they are referenced. However, FORTH words that are used must be defined.

If a reference is made to a state that is not defined, no error is generated until the state machine is run and there is an attempt to change to that state.

FORTH words defined by the application or by the system can be used within the state machine. Actually, the best way to think of the state machine words is like another programming language construct, rather like an extended DOCASE statement. Just as one must be careful how FORTH words are used in the DOCASE, one must be careful with the state machine.

The following FORTH words are provided to build a state machine:

STATE{ **(state number --)**

}STATE **(--)**

The **STATE{...}STATE** words bracket the definition of a single state within a state machine. STATE{ expects an integer which is the state number. If two states are defined with the same number, the second definition is used with no error messages.

ACTION{ **(truth flag --)**

}ACTION **(--)**

The **ACTION{...}ACTION** words bracket the FORTH code to be executed if the associated event occurs. ACTION{ expects a truth flag on the stack. If the truth flag is non-zero, the action list inside ACTION{ }ACTION is performed. If the truth flag was zero, the next event is checked. The events are defined by FORTH code that returns a flag to be used by the ACTION{ word. If the present event comparison results is a false condition (0), the next event is compared against the received event. There are high level words provided to perform the event comparisons needed. eg. FK_F1, TIMER (see EVENT RECOGNITION WORDS in Section 4).

Note: ACTION{...}ACTION is **only legal within** an STATE{...}STATE pair.

NEW_STATE **(state number --)**
Performs a change of the current state, i.e. the next event will cause activation of the Test Manager in this state. Another way to change the state is to modify the contents of the variable that contains the state. NEW_STATE leads to the execution of the new state's "STATE_INIT" code (if there is one and the supplied number differs from the one of the current state), while 1 NSTATE ! doesn't! **Note:** NEW_STATE is **only legal within** an ACTION{...}ACTION pair.

LOAD_RETURN_STATE (state number --)

RETURN_STATE (--)

LOAD_RETURN_STATE together with **RETURN_STATE** permits the user to program the equivalent of subroutine calls into the state machine. The procedure is as follows:

1. Load the state number of the state to which control is to be returned:
   `X LOAD_RETURN_STATE`
2. Load the subroutine state value using: `X NEW_STATE`.
3. At the end of each subroutine's "ACTION" parts you want to return execute: **RETURN_STATE.** If you want to return even if no event matches in the "subroutine", execute `RETURN_STATE` at the end of the subroutine's state too (prior }STATE).

Note: Nesting is not permitted, only one level is supported at this time.

STATE_INIT{ (state number --)

}STATE_INIT (--)

The **STATE_INIT{...}STATE_INIT** words bracket the definition of a sequence of FORTH words that are to be executed upon entry to the specified state. The STATE_INIT{ expects a single integer which is the state number. The STATE_INIT{...}STATE_INIT procedure may be used for any defined state but is not compulsory. Execution of the "STATE_INIT" code is done upon startup of a Testscenario (for state 0) and by changing the state with NEW_STATE.

SEQ{ (sequence number --)

}SEQ (--)

RUN_SEQ (sequence number --)

The **SEQ{...}SEQ** words bracket the definition of a sequence of FORTH words which are to be executed as part of the state machine action. SEQ{ expects an integer which is the sequence number. Up to 256 action sequences are supported with sequence numbers from 0 to 255. The **RUN_SEQ** word is used inside of an action to execute the specified sequence of FORTH words.

4. EVENT RECOGNITION

Event recognition words are placed after the STATE{ and before the ACTION{ words in the state construct. The current received event is compared to the desired event, and a truth flag of 1 is returned if the events match. A flag of 0 is returned if the events do not match. This flag will be checked by the ACTION{ word which determines whether or not the following action sequence is to be executed.

Event recognition words are provided for frames, timers, IPC-events and keyboard entry (text strings, and function and cursor keys). The following list summarizes these words:

?WAKEUP, ?RX_FRAME, ?COMMAND, ?RESPONSE, ?RX_MATCH, ?RX_SEARCH, ?ABORT, ?CRC_ERROR, ?TIMER, ?L1_EVENT, ?L2_SERVICE, ?PIGGYBACKED , ?TM-IPC, ?KEY, FK_F1, FK_F2, FK_F3, FK_F4, FK_F5, FK_F6, ?KEYBOARD, OTHER_EVENT

4.1. ?WAKEUP

The **?WAKEUP** word is used within the event field to detect the wakeup timer. Timer 34 is started for 100 milliseconds by **TM_RUN** when **WAKEUP_ON** is selected.

Format: `?WAKEUP`

4.2. ?COMMAND

The **?COMMAND** word can be used within the event field together with **?RX_FRAME** (see below) to check whether the received frame was a COMMAND.

Format: `?COMMAND`

4.3. ?RESPONSE

The **?RESPONSE** word can be used within the event field together with **?RX_FRAME** (see below) to check whether the received frame was a RESPONSE.

Format: `?RESPONSE`

Example:

```
3 STATE{  R#RR ?RX_FRAME ?COMMAND AND
          ACTION{
               " STATE 3 : ? RR Command frame received" BLU_BG DISPLAY
          }ACTION
`STATE
```

4.4. ?RX_FRAME

The ?RX_FRAME word is used within the event field to check whether the specified frame type has been received.

Format: FRAME-TYPE ?RX_FRAME

where FRAME-TYPE may be taken from the following list:

R#I	- Information frame
R#RR	- Receive Ready frame
R#RNR	- Receive Not Ready frame
R#REJ	- REJect frame
R#SABME	- Set Asynchronous Balanced Mode Extended frame
R#DISC	- DISConnect frame
R#UA	- Unnumbered Acknowledgement frame
R#DM	- Disconnected Mode frame
R#FRMR	- FRaMe Reject frame
R#UI	- Unnumbered Information frame
R-XID#	- Exchange Identity (XID) frame
R#INVFRM	- INValid FRaMe

Example:

```
10 STATE{ R#DISC ?RX_FRAME
          ACTION{
               S_UA 0 NEW_STATE
          }ACTION
}STATE
```

Note: for monitor applications, the user may specify the source of data as being the terminal or network. this is accomplished as follows:

```
From NETWORK  - R#I ?RX_FRAME PORT-ID @ TO_DCE_RX = AND

From TERMINAL - R#I ?RX_FRAME PORT-ID @ TO_DTE_RX = AND
```

4.5. ?RX_MATCH

The **?RX_MATCH** word is used within the event field to do a character comparison between a user defined key string and the target string within the information field of a received information frame. To accommodate "don't care" character positions, the question mark "?" (Hex 3F) may be used. The key string and target string must match exactly from the first character position of the received data field up to the end of the key string. The received target string may be longer than the key string.

Format: " key string" ?RX_MATCH
or X" hex string" ?RX_MATCH

Example:

```
2 STATE{  " AB1" ?RX_MATCH X" 414232" ?RX_MATCH OR
          ACTION{
               ." AB1 or AB2 received"
          }ACTION
}STATE
```

Note: For monitor applications, the user may specify the source of data as being the DCE or DTE. This is accomplished as follows:

```
" key string" ?RX_MATCH PORT-ID @ TO_DCE_RX = AND
                                  (or TO_DTE_RX)
```

4.6. ?RX_SEARCH

The **?RX_SEARCH** is the unanchored version of the **?RX_MATCH** word. This word is used within the event field to do a character comparison between a user defined key string and any portion of the target string. The key string may match starting at any character position on the information field of an I frame (target string).

Format: " key string" ?RX_SEARCH
or X" hex string" ?RX_SEARCH

Example:

```
5 STATE{  " logon"  ?RX_SEARCH
          ACTION{
               S:LOGON_RESP 6 NEW_STATE
          }ACTION
}STATE
```

Note: For monitor applications, the user may specify the source of data as being the DCE or DTE. This is accomplished as follows:

```
" key string" ?RX_SEARCH PORT-ID @ TO_DCE_RX = AND
                                   (or TO_DTE_RX)
```

4.7. ?ABORT

The **?ABORT** word is used within the event field to detect an abort frame.

Format: ?ABORT

Example:

```
5 STATE{  ?ABORT
          ACTION{
               ." Abort received"
          }ACTION
}STATE
```

4.8. ?CRC_ERROR

The **?CRC_ERROR** word is used within the event field to detect a CRC error.

Format: ?CRC_ERROR

Example:

```
6 STATE{  ?CRC_ERROR
          ACTION{
               ." CRC Error received"
          }ACTION
}STATE
```

4.9. ?TIMER

The **?TIMER** word is used within the event field to detect a timeout of the specified timer (started with START_TM_TIMER). **?TIMER** expects an integer value that will be used as the timer's identifier.

Format: TIMER-NUMBER ?TIMER

where TIMER-NUMBER is the number of the timer (started with START_TM_TIMER).

Example:

```
8 STATE{  21 ?TIMER
          ACTION{
               ." Timer 21 expired"
          }ACTION
}STATE
```

4.10. ?L1_EVENT

The **?L1_EVENT** word is used within the event field to check whether the specified layer1 event occured or not.

Format: `LAYER1-EVENT ?L1_EVENT`

where LAYER1-EVENT may be taken from the following list:

Basic Access:

R#ACTIVATE	– Layer 1 activated
R#DEACTIVATE	– Layer 1 deactivated
R#RI2ERR	– Info 2 received in unexpected state
R#LSTFRM	– Lost framing
R#RECOVERY	– Recovery for reported error
R#UNDEFINED	– Undefined error

Primary Rate Access :

R#REDALM	– Red alarm
R#YELALM	– Yellow alarm
R#LSTSIGL	– Lost of Signal Level
R#LSTPLOCK	– Lost of physical link lock
R#NORMAL	– Layer 1 back to normal
R#BPVL	– Bipolar violation
R#CRCR	– CRC error
R#MERR	– Multiframe error
R#FERR	– Frame error
R#CEPTSPEC	– CEPT special error

4.11. ?L2_SERVICE

The **?L2_SERVICE** word is used within the event field to check whether the specified confirmed service type has been received or not.

Format: `SERVICE-TYPE ?L2_SERVICE`

where SERVICE-TYPE may be taken from the following list:

DL-EST#IND	– Peer entity established the data link.
DL-EST#CONF	– Establishment of data link connection is confirmed.
DL-REL#IND	– Peer entity released the data link.
DL-REL#CONF	– Release of existing data link connection is confirmed.
DL-UDATA#IND	– Unacknowledged data is received.
DL-DATA#IND	– Acknowledged data is received.
DL-DATA#CONF	– Data packet is confirmed.

Note: **DL-DATA#CONF** is no primitive specified by CCITT but it is added to the K1197 LAPD software to make it possible to test received acknowledgements. DL-DATA#CONF is created always when a frame (I, RR, RNR or REJ) with an acknowledgement to previously sent I frames is received.

Since LAPD allows I frames to acknowledge received I frames (piggybacked acknowledgement), ?L2_SERVICE will return true (1) for DL-DATA#IND as well as for DL-DATA#CONF in case of reception of I frames that carries acknowledgement. Detection of a piggybacked acknowledgement can be done by using the word **?PIGGYBACKED** (see below).

Examples of the above mentioned cases:

Received

- I frame with ackn. => DL-DATA#IND and ?PIGGYBACKED true
  (also DL-DATA#CONF and ?PIGGYBACKED true)
- I frame without ackn. => DL-DATA#IND and ?PIGGYBACKED false
- RR with ackn. => DL-DATA#CONF and ?PIGGYBACKED false

Example:

```
10 STATE{ DL-DATA#CONF ?L2_SERVICE
          ACTION{
               DL-RELEASE 11 NEW_STATE
          }ACTION
}STATE
```

4.12. ?PIGGYBACKED

The **?PIGGYBACKED** word is used within the event field to check whether an I-frame that carries acknowledgement for previously sent I frame(s) is received. ?PIGGYBACKED can be used in conjunction with ?L2_SERVICE to check more detailed for the type of DL-DATA#IND or DL-DATA#CONF. (see the examples below and above)

Format: ?PIGGYBACKED

Example 1:

```
0 STATE{  DL-DATA#IND ?L2_SERVICE
          ACTION{
               ?PIGGYBACKED
               IF    ." I-frame with ACKNOWLEDGEMENT"
               ELSE ." I-frame without ACKNOWLEDGEMENT"
               ENDIF
          }ACTION
}STATE
```

Example 2:

```
0 STATE{  DL-DATA#CONF ?L2_SERVICE ?PIGGYBACKED AND
          ACTION{
               ." I-frame with ACKNOWLEDGEMENT"
          }ACTION

          DL-DATA#CONF ?L2_SERVICE
          ACTION{
               ." ACKNOWLEDGEMENT"
          }ACTION
}STATE
```

4.13. ?TM-IPC

The ?TM-IPC word is used within the event field to check whether the specified message is received from the specified Test Manager or not.

Format: EXPECTED-EVENT EXPECTED-SENDER ?TM-IPC

where EXPECTED-EVENT may be either a special value (the self defined "message") except -1 or **ANY#IPC-EVENT** (any message will match) and EXPECTED-SENDER may be the senders Test Manager number (0..7) or **ANY#TM** (messages from any TM's will match).

Example:

```
0 STATE{  5 3 ?TM-IPC
          ACTION{
               ." MESSAGE #5 RECEIVED FROM TM#3"
          }ACTION
}STATE
```

4.14. ?KEY

The **?KEY** word is used within the event field to detect function or cursor keys. ?KEY expects the constant value identifying the key. These values are listed under **EXTRACT_KEY** in the FORTH PROGRAMMING User Manual.

Format: KEY-IDENTIFIER ?KEY

where KEY-IDENTIFIER is the key identifier (see the list under EXTRACT_KEY in the FORTH PROGRAMMING User Manual).

Example:

```
9 STATE{  CHOME ?KEY
          ACTION{
               ." HOME key pressed"
          }ACTION
}STATE
```

Note: Only the currently active Test Manager is able to detect function or cursor keys.

By pressing the cursor right key, by default the first menu will be opened. If this function should be suppressed (e.g. if the test manager uses the cursor right key for self defined actions) the following two words are available to control the opening of the menu system.

MENU_OFF (--)
Suppresses function of cursor right key to open the menu system.

MENU_ON (--)
Allows opening of the menu system by pressing the cursor right key (default).

4.15. FK_F1..FK_F6

Six function keys have been made available for assignment by the user. The words that can be used in place of ?KEY within the event field are:

FK_F1	– Detects function key f1
FK_F2	– Detects function key f2
FK_F3	– Detects function key f3
FK_F4	– Detects function key f4
FK_F5	– Detects function key f5
FK_F6	– Detects function key f6

Example:

```
9 STATE{  FK_F1
          ACTION{
               ." f1 key pressed"
          }ACTION
}STATE
```

Note: Only the currently active Test Manager is able to detect function or cursor keys.

4.16. ?KEYBOARD

The Test Manager event word **?KEYBOARD** has been added to allow for entry of character strings or numerical values into the Test Manager states. States which contain **?KEYBOARD** will stop processing of keyboard entries by the FORTH interpreter and will pass the character string to the Test Manager action instead. The **?KEYBOARD** word will return a flag of 1 when it detects keyboard entry terminated with the return key. The character string can be accessed in the action by using HERE to get the address of the character string. The length of the string will be in the first byte. The word **EXPECT_NUM** can be used to convert the character entry into a numerical value. **EXPECT_NUM** will return the value and a flag on the stack. The flag will be 1 if the number conversion was valid and will be 0 if the entry could not be converted to a numerical value.

Format: ?KEYBOARD

Example 1: (Usage for character string entry)

```
1 STATE{  ?KEYBOARD              ( check for keyboard entry )
          ACTION{
               HERE              ( get address of string )
               COUNT TYPE        ( this example will display it )
          }ACTION
}STATE
```

Example 2: (Usage for numerical value entry)

```
2 STATE{ ?KEYBOARD
         ACTION{
              EXPECT_NUM          ( get value and flag on stack )
              IF
                   ." value = " . ( display value )
              ELSE
                   DROP ." Invalid entry"
              THEN
         }ACTION
}STATE
```

4.17. OTHER_EVENT

The **OTHER_EVENT** word will return a true flag that can be used in the event field to always execute the actions. This can be used to detect any event which has not been explicitly programmed into an event field. The actual decoded event value will be found in the variable **EVENT-TYPE** and the user may wish to display or check it by using **EVENT-TYPE ?** or **EVENT-TYPE @** . The event types are identified by the constants **FRAME#**, **LAYER1#**, **FUNCTION-KEY#**, **STRING-ENTRY#**, **TIME-OUT#**, **L1-TIME-OUT#**, **TM-TIME-OUT#**, **TM-IPC#** and **UNKNOWN-EVENT#**.

See the ISDN EMULATION/SIMULATION Manual for detailed decoding of events.

Note: **OTHER_EVENT** must be the last event detection in the STATE{...}STATE.

4.18. SELF DEFINED EVENT RECOGNITION

With advanced knowledge of the LAYER 2 EMULATION it is possible to define your own event recognition. Since **ACTION{** expects a truth flag on the FORTH stack, any sequence of code which leaves a flag can be used for event recognition.

The actual decoded event value will be found in the variable **EVENT-TYPE** and the user can check it by using **EVENT-TYPE @** . The event types are identified by the constants **FRAME#**, **LAYER1#**, **FUNCTION-KEY#**, **STRING-ENTRY#**, **TIME-OUT#**, **L1-TIME-OUT#**, **TM-TIME-OUT#**, **TM-IPC#** and **UNKNOWN-EVENT#**. All necessary information for detailed event decoding is included in the "VARIABLES" part of the ISDN EMULATION/SIMULATION Manual.

Note: The variables described in the "VARIABLES" part of the ISDN EMULATION/SIMULATION Manual are valid **only** if the L.2 and ML/ASP state machines are turned **OFF** !

4.19. SELF DEFINED EVENT GENERATION

A mechanism is available to generate an event within a testscript which can be recognized in another TM-state. This is useful to abstract a teststep from a "higher layer" and to implement testscenarios strictly according to SDL-diagrams. The following words are available for this feature.

SET_EVENT_FLAG **(--)**
generates an event which will be recognized by ?TEST_EVENT.

?TEST_EVENT **(-- flag)**
is the Test manager recognition word which will return a true flag if an event has been generated by SET_EVENT_FLAG before.

<TEST_EVENT_ACTION> **(--)**
DOER word which will be executed by the occurence of the TEST event.

Example:
(The constants #ALERT and #CONNECT and the variables UT-TYPE and AUTO-MODE must be defined.)

```
: UT_CONN ( -- )                          ( generation of event for CONNECT )
     MAKE <TEST_EVENT_ACTION>
          #CONNECT UT-TYPE !
     ;AND
     SET_EVENT_FLAG
;

: UT_ALERT ( -- )                         ( generation of event for ALERT )
     MAKE <TEST_EVENT_ACTION>
          #ALERT UT-TYPE !
     ;AND
     SET_EVENT_FLAG
;

: TU_SETUP ( -- )                         ( decision between ALERT and CONNECT )
     AUTO-MODE @
     IF
          UT_CONN
     ELSE
          UT_ALERT
     ENDIF
;
```

```
( The generated event can be detected by the following two testmanager )
( recognition words. )

: ?UT_ALERT ( -- flag )
     ?TEST_EVENT
     UT-TYPE @ #ALERT =
     AND
;

: ?UT_CONN ( -- flag )
     ?TEST_EVENT
     UT-TYPE @ #CONNECT =
     AND
;

( Testscenario example to send either an ALERT or a CONNECT message after an )
( incoming SETUP. )

0 STATE{ M#SETUP 1 ?L3_VALID_MSG
          ACTION{
               TU_SETUP
               6 NEW_STATE
          }ACTION
}STATE

6 STATE{ ?UT_ALERT
          ACTION{
               " ALERT" SEND_MESSAGE
          }ACTION

          ?UT_CONN
          ACTION{
               " CONNECT" SEND_MESSAGE
          }ACTION
}STATE
```

5. ACTION WORDS

Any of the commands defined in the ISDN Monitor or Emulator can be executed as part of an action sequence. The user may also use any FORTH words that are in the dictionary or compile new words into the dictionary. The P." group of words defined in the K1197 Users Manual can be used for printer output. The T." Trace report group of words can be used to combine reports from Test Manager actions with the data display.

5.1. DISPLAY

5.1.1. COMMAND LINE DISPLAY

This word permits the user to generate comments, prompts, traces, etc. in the command window.

Format: `"TEXT STRING" ATTRIBUTE DISPLAY`

where the text string must be entered between two quote symbols. Note that a space character must follow the first quote. ATTRIBUTE may be one or a combination (via the OR or + operators) of the following:

WHI_FG	– White Foreground
GRN_FG	– Green Foreground
YEL_FG	– Yellow Foreground
CYA_FG	– Cyan Foreground
BLU_BG	– Blue Background
RED_BG	– Red Background
MAG_BG	– Magenta Background
BLK_BG	– Black Background
REV_VIDEO	– Reverse Video

The normal display words from FORTH can be used to generate display in the command window too (see the K1197 FORTH Programming Language Manual).

5.1.2. STATUS LINE DISPLAY

The user can control the display to the Status Window (lines 2 and 3 of the screen display). This can be done by using the following procedure.

```
0 HEADER-STATE !                ( Turn status display off )
SL-WINDOW @ POP_WINDOW          ( Open and display user status lines )
CLEAR_TEXT                      ( Clear text from window )
WHI_FG PAINT                    ( Clear color from window )
0 10 THERE                      ( Position in status window )
W." Display in status window"   ( Display text in status window )
CLOSE_WINDOW                    ( Close status window )
```

5.2. FUNCTION KEYS

One set of function keys is reserved for use by the test manager. F7 is "TM_RUN", which starts Test Manager execution, while F8 ("MON KEYS") switches back to the monitor function key set. The keys F1–F6 can be used by the test script itself. The labels for F1..F6 are local for each Test Manager. Only the labels of the currently active Test Manager are displayed. To display a specific string in a function key's label field, use **=TM_KEY .**

=TM_KEY **(string address / key number --)**
Sets the label for the specified **key number** to the string supplied at **string address**, where **key number** may be taken from the following list:

1 – function key 1
2 – function key 2
3 – function key 3
4 – function key 4
5 – function key 5
6 – function key 6

5.3. LAPSE TIMERS

There are 256 interval timers that can be started and read at any time.

START_LAPSE_TIMER **(0..255 --)**
This command starts the specified timer.

MINUTES_ELAPSED **(0..255 -- minutes)**
This command will check the specified timer and return the number of minutes elapsed since the timer was started.

SECONDS_ELAPSED **(0..255 -- seconds)**
This command will check the specified timer and return the number of seconds elapsed since the timer was started.

MILLISECONDS_ELAPSED **(0..255 -- milliseconds)**
This command will check the specified timer and return the number of milliseconds elapsed since the timer was started.

5.4. INTERVAL TIMERS

There are 35 timers available for each Test Manager. Use the **?TIMER** event to detect an interval timeout within the Test Manager state machine. Each Test Manager can start and stop its own timers (timer id values are local to each Test Manager) with the following commands:

START_TM_TIMER **(timer id / value --)**
The timers are started by specifying the timer id and the timeout interval in tenths of seconds.
Example: `1 300 START_TM_TIMER` – will start timer 1 for 30 seconds.

STOP_TM_TIMER **(timer id --)**
Stops the specified timer for the currently active Test Manager
Example: `90 STOP_TM_TIMER` – will stop timer 90.

STOP_ALL_TIMERS **(--)**
Stops all timers for the current Test Manager.

5.5. IPC MESSAGE SENDING

Test Managers are able to communicate with each other via an IPC mechanism. The procedure is as follows:

Sending an IPC message is done using the word TM_IPC_SEND. When the active Test Manager issues this command, the destination Test Manager is (immediately) executed in its current state. After execution of the TM_IPC_SEND, the caller's test step is continued. An error message is generated if the destination Test Manager is not ready or if it does not excpect the IPC event (via ?TM_IPC) in its present state.

TM_IPC_SEND **(event / receiver --)**
Will immediately execute the receiver Test Manager in it's present state, then returns to the word following the call.

event may be a special (user definable) value except -1.
receiver is the number of the destination Test Manager (0..7).

Note: No range check is done for receiver, so ensure that receiver=0..7. The value of EVENT-TYPE will be destroyed when calling TM_IPC_SEND.

5.6. BUFFER CREATION

Up to 256 buffers can be created and used by the Test Manager. The buffers can be created any size desired, however, the ISDN software will not transmit more than 260 bytes. The buffers are specified by numbers 0 to 255.

BUFFER STRUCTURE

The buffer structure consists of 4 bytes with values of zero, 2 bytes containing the 16 bit length, followed by the text bytes.

BUFFER

(number -- addr of buffer)
The BUFFER command is used to specify a buffer and will return its starting address.
Format: `0 BUFFER`

FILE->BUFFER

(" filename" / number --)
The FILE->BUFFER command is used to transfer a text file into the buffer specified. The file is transferred into the buffer until an ASCII CTRL P (displayed as a Red Pilcrow character) is found or the file is finished.
Format: `" FILENAME" 0 FILE->BUFFER`

STRING->BUFFER

(" string" / number --)
The STRING->BUFFER command is used to load a quote string into the specified buffer. The length is limited to 255 bytes.
Format:
```
" TEXT STRING" 1 STRING->BUFFER
X" 0100100100434445" 2 STRING->BUFFER
```

BUFFER_LENGTH

(number -- length of the buffer)
The BUFFER_LENGTH command will return the current size of the specified buffer.

6. TRACE REPORT COMMANDS

Trace reports can be created and displayed together with the protocol report lines. These trace reports can also be saved in capture ram and in disk recordings. The words that are used for creating and displaying trace reports are: TEMIT, T.TYPE, T." ", T., T.H, TCOLOR, TCR.

TEMIT **(value --)**
Inserts the character value passed to it on the FORTH stack into the current position of the trace report buffer. The TEMIT operates for the trace reports like EMIT does for command lines.

T.TYPE **(addr / count --)**
Inserts the character string starting at address with a length of count into the current position of the trace report buffer. The T.TYPE operates for trace reports like TYPE does for command lines.

T." **(--)**
Inserts a character string into the trace report buffer. T." is used in the form: `T." text string"` where, the terminating delimiter is a " (quote). The T." operates for trace reports like ." does for command lines.

T. **(value --)**
Performs a binary to ASCII conversion on the value that is on the FORTH stack and inserts the result followed by one space into the trace report buffer. BASE is used to determine the number base for the conversion with decimal being the default. The T. operates for trace reports like . does for command lines.

T.H **(value --)**
Performs a binary to ASCII conversion using hex base on the value that is on the FORTH stack and inserts the result followed by one space into the trace report buffer. The T.H operates for trace reports like .H does for command lines.

TCR **(--)**
Performs a line for the trace report. The display of the trace is done when the TCR is executed. The TCR operates for trace reports like the CR does for command lines.

TCOLOR **(color --)**
Allows changing of the color of the trace report lines. The colors that can be used are the same as described in Section 5.1.1

RTRACE **({YES|NO} --)**
Selects report trace display on (YES RTRACE) or off (NO RTRACE).

CTRACE **({YES|NO} --)**
Selects capture trace on (YES CTRACE) or off (NO CTRACE).

DTRACE **({YES|NO} --)**
Selects disk recording trace on (YES DTRACE) or off (NO DTRACE).

STATE-TRACE **(-- addr)**
The variable STATE-TRACE controls, whether state transition will be reported (1) or not (0).

7. TEST MANAGER ACTIVATION

The Test Manager is activated for frame events, keyboard events, timer events or through IPC events. Monitoring and Triggering, as well as executing the L.2 and ML/ASP state machines is done before activating the Test Manager. For frame events the following activation strategy is used:

MONITOR mode:
The Test Manager is activated for each incoming frame.

TE emulation:
The Test Manager is activated for all DLCI's in case of Management frames (TEI=127,SAPI=63), for all DLCI's whose SAPI matches the incoming frame's SAPI in case of broadcasting frames (TEI=127). For other frames the corresponding DLCI is searched and the Test Manager is activated for this specific DLCI.

NT emulation:
Activation works the same way as in TE emulation, with the exception that the Test Manager is activated only once for Management frames (only 1 ASP module for NT).

8. LAYER 3 DECODER VARIABLES

The following FORTH variables are provided by the Layer 3 Message Decoder to provide information about the most recently received message. This information includes the protocol discriminator, call reference and message type as well as detailed information about detected errors.

$MSG_ERROR **(-- addr)**
This variable contains an error code identifying the error that occurred when decoding the last received message. It may have one of the following values which are defined as constants (Decimal values are given in brackets):

Error Code	Meaning
NO-ERROR#	(0) No error was detected.
RANGE-ERR#	(1) Information element exceeds message range
SHIFT-ERR#	(2) Shift after Single Shift detected
INV-CALLREF#	(3) Call reference has invalid format
INV-DISCR#	(4) Unknown protocol discriminator received
SHORT-MSG#	(5) Received message is too short
NO-MSG#	(6) No message received

Note that depending on the value of $MSG_ERROR all or some of the other decoder variables may or may not contain valid information (eg. when the protocol discriminator was unknown there will be no valid entry in $MSG_TYPE).

$MSG_DISCR **(-- addr)**
Contains the Protocol Discriminator received within the last message.

$MSG_CR_LEN **(-- addr)**
Contains the length of the Call Reference extracted from the last received message.

$MSG_CR_FLAG **(-- addr)**
Contains the value of the Call Reference Flag. Note that this flag is completely removed from the Call Reference Value and may be only examined from this variable.

$MSG_CR_VALUE **(-- addr)**
Contains the Call Reference Value as an normal 32-bit integer number. The actual length of the call reference and the value of the call reference flag can be obtained in $MSG_CR_LEN and $MSG_CR_FLAG respectively.

$MSG_TYPE **(-- addr)**
Contains the Message code identifying the type of the last received message. This is the actual bit value used in coding the message.

$MSG_ID **(-- addr)**
Contains the Message identification assigned to the type of the last received message. This is the value that the user provides to identify a particular message type.

9. LAYER 3 DECODER COMMANDS

The Layer 3 software generates an event on the reception of a Layer 3 Message. The following FORTH commands provide the user with the ability to determine the type and content of received Layer 3 events, such that the appropriate action can be executed:

?L3_MSG	**(MSG index / ... / MSG index / n -- flag)**
?L3_MATCH	**(string -- flag)**
?L3_SEARCH	**(string -- flag)**
?L3_VALID_MSG	**(MSG index / ... / MSG index / n -- flag)**
?L3_IECOUNT	**(IE index -- count)**
?L3_IE	**(IE index / n -- flag)**
?L3_MAND_IE	**(MSG index -- flag)**
?L3_OPT_IE	**(MSG index -- flag)**
?L3_MAND_IE_MISSING	**(MSG index -- flag)**
?L3_UNREC_IE	**(MSG index -- flag)**
?L3_MAND_IE_ERR	**(MSG index -- flag)**
?L3_OPT_IE_ERR	**(MSG index -- flag)**
?L3_MAND_IE_ORDER	**(MSG index -- flag)**
?L3_OPT_IE_ORDER	**(MSG index -- flag)**
GET_IES	**(-- IE code 1 / ... / IE code n / n | 0)**
?L3_UNEXP_IE	**(MSG index -- flag)**
?L3_OCTET	**(IE index / n / octet index -- flag)**
?L3_IE_ORDER	**(-- flag)**
?L3_IE_ERROR	**(IE index -- flag)**

?L3_MSG **(MSG-id / .. / MSG-id / n -- t:f)**

Checks if a particular Layer 3 message has been received. A list of user specified Message identifications (which represent particular Layer 3 messages) are compared to the Message identification decoded for the received event. If the decoded value is equivalent to a value from the Message identification list, a true flag is placed on the stack otherwise a false flag is left. Consequently decisions based on the message or messages received can be made within the Simulation (ie. within a state definition of the Test Manager). n is the number of supplied Message Identifications.

Example: Check for the reception of CALL SENT or ALERT messages :

```
...
M#CALL_SENT M#ALERT 2 ?L3_MSG
ACTION{
     < perform action required if CALL SENT or ALERT messages
       received  >
}ACTION
...
```

?L3_MATCH **(string -- t:f)**

Permits character comparison between a user defined key string and a target string within a Layer 3 message. If the key string and target string match exactly from the first character of the received message to the end of the key string a true flag is placed on the stack. The received target string may be longer than the key string. "Don't care" character positions within the key string can be specified with a question mark, "?" (Hex 3F).

Example: Format for string matching within messages.

```
" key string" ?L3_MATCH        ( or can use   X" hex string" )
ACTION}
     < perform action required if string match occurred >
     < .... >
}ACTION
...
```

?L3_SEARCH **(string -- t:f)**

The unanchored version of the ?L3_MATCH command. This command permits character comparison between a user defined key string and any portion of a target string within a Layer 3 message. If the key string matches with any part of the target a true flag is placed on the stack. As with ?L3_MATCH, "don't care" character positions are indicated with a question mark, "?" (Hex 3F).

?L3_VALID_MSG **(MSD-identifier / .. / MSG-identifier / n -- t:f)**

Returns a true flag if one of the specified messages was last one to have been received, and was coded correctly. The number of specified messages must also be given.

?L3_IECOUNT **(IE-identifier -- count)**

Returns a value to the stack which indicates the number of times the specified Information Element was found in the received message.

Example: Check the presence of the Cannel Id IE :

```
...
I#CHANNEL_ID ?L3_IECOUNT
IF
     < Channel Id was present in the received message >
     < now process it,eg.decode it >
ELSE
     < CHI was not present >
THEN
...
```

?L3_IE **(IE-identifier / n -- t:f)**
Searches for the n-th occurrence of a specified Information Element and if it is found returns a true flag on the stack. However, prior is the true flag being placed on the stack the Information Element is decoded and the resulting information is placed into the associated Information Element parameters.

Example: Try to decode the 3-rd CAUSE IE.

```
I#CAUSE 3 ?L3_IE
IF
     < perform action required if 3rd CAUSE-IE found>
ELSE
     < perform action required if 3rd CAUSE-IE not found>
THEN
...
```

?L3_IE_ORDER **(-- t:f)**
Returns a true flag to the stack if all Information Elements occur in correct order within the received message. If multiple IEs of one type occur within the message, they must be grouped together.

?L3_MAND_IE **(MSG-identifier -- t:f)**
Returns a true flag to the stack if all mandatory Information Elements for the specified message type were found.

?L3_OPT_IE **(MSG-identifier -- t:f)**
Returns a true flag to the stack if at least one optional Information Element was found for the specified message type.

?L3_MAND_IE_MISSING **(MSG-identifier -- t:f)**
Searches missing mandatory IEs and if there are one or more IEs missing the word will return a true flag. The user can get the IEs by calling the word GET_IEs.

?L3_UNREC_IE **(-- t:f)**
Searches unrecognized IEs in the last decoded message. If there are one or more unrecognized IEs the word will return a true flag. GET_IEs will return these IEs.

?L3_MAND_IE_ERR **(MSG-identifier -- t:f)**
Searches for mandatory IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_OPT_IE_ERR **(MSG-identifier -- t:f)**
Searches for optional IE errors and returns a true flag if it has found one or more. GET_IEs will return these IEs.

?L3_MAND_IE_ORDER **(MSG-identifier -- t:f)**
Checks if all mandatory IEs ar in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

?L3_OPT_IE_ORDER **(MSG-identifier -- t:f)**
Checks if all optional IEs are in correct order and returns a true flag if there is a sequence error. GET_IEs will return the wrong placed IEs.

GET_IES **(-- IE-code1/.../IE-code n/n | 0)**
Returns the wrong IEs from the last called check word or a false flag if there was no error detected.

?L3_UNEXP_IE **(MSG-identifier -- t:f)**
Returns a true flag to the stack if at least one unexpected Information Element was found for the specified message type. An Information Element is considered to be unexpected when it is neither mandatory nor optional, or when it is mandatory or optional but illegaly occurs multiple times within a message.

Example: Check IE occurrences within a received message:

```
...
M#SETUP ?L3_UNEXP_IE      ( Check for unexpected IEs)
?L3_IE_ORDER 0= OR        ( or IE order )
IF
     < handle Setup messages containing unexpected or badly
       placed IEs >
ELSE
     M#SETUP ?L3_MAND_IE 0=
     IF
          < handle Setup messages with missing mandatory IEs >
     ELSE
          < handle Setup messages with all required IEs >
          M#SETUP ?L3_OPT_IE       (Check for optional IEs)
          IF
               < handle optional IEs >
          THEN
-THEN
-THEN
...
```

?L3_OCTET **(IE-identifier / n_th occurrence of IE / OCTET-identifier -- t:f)**
Returns a true flag to the stack if the nominated octet was found within the specified IE.

Example: Check if octet_4 is present in the first Calling Number IE of the received message:

```
...
I#CALLING_NUM 1 OCTET_4 ?L3_OCTET
IF
     < process address field >
THEN
...
```

?L3_IE_ERROR **(IE-identifier -- d)**

Returns an error code identifying the error that occurred while decoding nominated Information Elements. The following error codes are defined as constants (Decimal values are given in brackets):

Error Code	Meaning
NO-IE-ERROR#	(0) No error was detected.
IE-TOO-SHORT#	(1) Information element content too short.
IE-TOO-LONG#	(2) Information element content too long.
UNEXP-EXTEND#	(3) Unexpected extension found.
EXP-EXTEND#	(4) Expected extension not found

Example: Print the error code after decoding has failed:

```
...
I#CAUSE 1 ?L3_IE                ( if decoding of IE is possible )
IF
     I#CAUSE ?L3_IE_ERROR                ( get error code )
     ?DUP
     IF                          ( it decoding error occurred )
          T." IE-ERROR " T. TCR     ( display error message )
     THEN
THEN
...
```

10. ACCESSING INFORMATION ELEMENT FIELD VALUES

Access of Information Element parameters requires a sequence of preceeding checks to make sure that:

- the Information Element was present in the received message
- the n-th occurrence of the Information Element was decoded properly
- the octet containing the addressed parameters was present in the Information Element

The command **?L3_IECOUNT** indicates the number of times the specified Information Element was found within the received message. Note that this number may be used to enter a loop that handles all Information Elements of one type. The complete decoding of an Information Element is performed when the command **?L3_IE** is used.

?L3_IE decodes the n-th occurrence of one Information Element type and places the set of decoded information into a data structure called the Information Element parameter list. Note that the Information Element parameter list contains only one set of parameter values for each Information Element type at one time.

Prior to accessing a parameter the command **?L3_OCTET** should be used to determine whether the octet associated with the parameter value was found in the Information Element or not. A parameter value is supposed to be only valid (ie. extracted from the Information Element) when **?L3_OCTET** indicates a positive result.

A parameter value may be obtained by using the FORTH constant ***DEC** which places the base address of the Information Element parameter list on the stack. This is followed by an offset adding command, which generates the address of the required parameter value. The FORTH command @ (fetch) may be used to place the specified parameter value on the stack, unless the parameter is a string parameter (eg. addresses).

Example: Access of parameters in the Calling Number IE :

```
I#CALLING_NUM 1 ?L3_IE              (decode first CALLING Number IE)
IF
     *DEC  ->CLGN_NUMBER_TYPE  @        (process address type)
     ...
     I#CALLING_NUM 1 OCTET_4 ?L3_OCTET       (check if octet 4 is
                                              included)
     IF
          *DEC  ->CLGN_NUMBER       (process address string)
          ...
     THEN
ELSE
     < handle decoding error >
THEN
```

Complete information for access of the Information Element parameter list including relevant octet identifier for all Information Elements, selector names associated with each Information Element parameter as well as a complete list of predefined constants for each standardized parameter value are given in the ISDN NATIONAL VARIANT's Manual.

11. TEST SCRIPT EXAMPLES

11.1. TEST0.F

```
(-------------------------------------------------------------------------)
( File      : TEST0.F                                                     )
(-------------------------------------------------------------------------)
( Purpose   : Example for using keyboard entry within Test Manager scenarios  )
(-------------------------------------------------------------------------)

TCLR                                ( Clear the state machine )
MONITOR                             ( Turn on monitor mode )

0 STATE_INIT{

     ( print a prompt message in the command windows: )
     CR ." Press F1 for Character string entry, F2 for Numeric entry " CR

     ( and define the Function key labels: )
     " CHARACTER"   1 =TM_KEY
     "  NUMERIC "   2 =TM_KEY

}STATE_INIT

0 STATE{
     FK_F1                          ( Method 1: Detect function key 1 )
     ACTION{
          ." ENTER CHARACTER STRING" CR
          TM_DEF_KEYS
          1 NEW_STATE
     }ACTION

     UF2 ?KEY                       ( Method 2: Detect function key 2 )
     ACTION{
          ." ENTER NUMBER VALUE" CR
          TM_DEF_KEYS
          2 NEW_STATE
     }ACTION
}STATE

1 STATE{
     ?KEYBOARD                      ( Keyboard Entry ? )
          ACTION{
               HERE COUNT T.TYPE TCR
                                    ( Echo the string input )
               0 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE{
     ?KEYBOARD                    ( Keyboard Entry ? )
          ACTION{
               EXPECT_NUM         ( Is the entry a number ? )
               IF
                    T." VALUE = " T. TCR
                                  ( Echo the numeric entry )
               ELSE
                    DROP
                    T." Invalid number entry" TCR
               THEN
               0 NEW_STATE
          }ACTION
}STATE
```

11.2. TEST1.F

```
(-------------------------------------------------------------------------)
( File      : TEST1.F                                                     )
( Function : Testing of layer 2 without using L2 - Services. Establishes  )
(            a connection and after getting a link the test scenario      )
(            starts sending I frames. After acceptance of the I frame a new )
(            one will be send. The link will be disconnected when 5 I frames )
(            have been send.                                              )
(-------------------------------------------------------------------------)
( Note      : This example shows how the test manager in user emulation   )
(            mode could be used without confirmed L2 - Services. For that )
(            need it is necessary to turn the layer 2 or management layer )
(            state machine on or off. {L2_ON, L2_OFF, ML_ON, ML_OFF}.     )
(-------------------------------------------------------------------------)

#IFNOTDEF COUNTER1
     0 VARIABLE COUNTER1
#ENDIF

TCLR                                ( Clear the state machine )
MONITOR                             ( Turn on monitor mode )


" Hello, this is SIEMENS K1197 !" MAKE_DATA1
1 PKT

0 STATE_INIT{
     0 COUNTER1  !                  ( Initialize variable COUNTER1)
     0 NS
     0 NR                           ( Initialize NR and NS)
     P0 F0                          ( Initialize poll- and final-bit)
     " YOU ARE THE USER " MAG_BG DISPLAY
     ."Press F1 to start test scenario " CR
     BLU_BG TCOLOR
     "   START  "  1 =TM_KEY
     L2_ON
}STATE_INIT
```

```
0 STATE{
     FK_F1                          ( Detect function key 1 )
          ACTION{
               T." Test scenario starts " TCR
               ML_OFF               ( Turn management layer state machine off)
     #IF PH_I/F #BRI = #IS_TRUE     ( This is part compiled only with Basic
                                        rate interface, not primary rate )
               MDL_ASSIGN_I         ( Indicate the need for any TEI )
               R#UI ?RX_FRAME       ( TEI assignment was successful )
               IF
     #ENDIF
                    ML_EXEC         ( Execute layer 2 state machine )
                    ML_ON           ( Turn management layer state machine on )
                    L2_OFF          ( Turn layer 2 state machine off )
                    P1              ( set poll-bit )
                    S_SABME    ( Send a SABME outside ctrl of state machine )
                    L2_EXEC         ( Execute layer 2 state machine )
                    1 NEW_STATE
     #IF PH_I/F #BRI = #IS_TRUE
               THEN
     #ENDIF
          }ACTION
}STATE

1 STATE{
     R#UA ?RX_FRAME
          ACTION{
               T." Link is established now " TCR
               0 DUP NR NS          ( Initialize NR and NS )
               P0 F0                ( Reset poll- and final-bit )
               S_I                  ( Send I frame outside crtl of state
                                        machine )
               1_COUNTER1 +!        ( Increment counter for sending loop )
               2 NEW STATE
          }ACTION

     OTHER_EVENT
          ACTION{
               L2_EXEC              ( Execute layer 2 state machine )
          }ACTION
}STATE
```

```
2 STATE{
     R#RR ?RX_FRAME
          ACTION{
               T." I frame no. " COUNTER @ T. T." send and accepted." TCR
               COUNTER1 @ 5 =       ( Termination for sending loop )
               IF
                    T." ALL I frames accepted ---> release the link " TCR
                    P1              ( Set poll-bit )
                    S_DISC          ( Send DISC outside ctrl of state machine )
                    3 NEW_STATE
               ELSE
                    0 NR            ( Reset NR )
                    COUNTER1 @ NS   ( Reset NS )
                    S_I             ( Send frame outside ctrl of state
                                         machine )
                    1 COUNTER1 +!   ( Increment counter for sending loop )
               THEN
          }ACTION

     OTHER_EVENT
          ACTION{
               L2_EXEC              ( Execute layer 2 state machine )
          }ACTION
}STATE

3 STATE{
     R#UA ?RX_FRAME
          ACTION{
               L2_ON                ( Turn layer 2 state machine on )
               T." TEST DONE " TCR
               0 NR 0 NS            ( Reinitialize NR and NS )
               P0 F0                ( Reinitialize poll- and final-bit )
               0 NEW_STATE
          }ACTION
}STATE
```

11.3. TEST2.F

```
(-------------------------------------------------------------------------)
( File      : TEST2.F                                                     )
(-------------------------------------------------------------------------)
( Function : Testing of the L2 - Service. Establishes a connection and when  )
(            the test scenario gets a establish confirmation              )
(            " DL-EST#CONF ?L2_SERVICE " it starts to send I frames.      )
(            The data transfer is confirmed, that means every new I frame )
(            will be send after the confirmation of the last send         )
(            I frame. The connection will be disconnected with            )
(            the confirmation of 5'th I frame                             )
(-------------------------------------------------------------------------)
( Note      : Testscript is for user emulation only.                      )

#IFNOTDEF COUNTER1
     0 VARIABLE COUNTER1
#ENDIF

TCLR
MONITOR                              ( Turn on monitor mode )

" Welcome to the machine "                    MAKE_DATA1
" It's one o'clock and time for lunch "       MAKE_DATA2
" Bam di dam di damdam "                      MAKE_DATA3
" And the sun has been out for a couple of hours "     MAKE_DATA4
" ..............."     MAKE_DATA5

SEQ{                                 ( define a sequence to send altering
                                         data packets )
     COUNTER1 @ 5 MOD 1+ PKT         ( select the data packet )
     DL_DATA                         ( send it )
     1 COUNTER1 +!
}SEQ

0 STATE_INIT{
     ." Please poke F1 to start the test scenario ! " CR
     "     START " 1 =TM_KEY
     0 COUNTER1 !                    ( Initialize COUNTER1 )
     BLU_BG TCOLOR
     L2_ON                           ( Turn layer 2 state machine on )
}STATE_INIT

0 STATE{
     FK_F1
          ACTION{
               DL_ESTABLISH          ( Establish the link  )
               T." Establish the connection. " TCR
               1 NEW_STATE
          }ACTION
}STATE
```

```
1 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

1 STATE{
     DL-EST#CONF ?L2_SERVICE
          ACTION{
               T." Establish confirmation --- link is existing. "TCR
               0 RUN_SEQ            ( Remember: that sequence sends a packet )
               2 NEW_STATE
          }ACTION
}STATE

2 STATE{
     DL-DATA#CONF ?L2_SERVICE
          ACTION{
               T." DATA PACKET NO. " COUNTER1 @ T. T." CONFIRMED " TCR
               COUNTER1 @ 5 =
               IF
                    DL_RELEASE
               T." ALL 5 PACKETS CONFIRMED -> RELEASE THE CONNECTION " TCR
                    3 NEW_STATE
               ELSE
                    0 RUN_SEQ       ( Remember: that sequence sends a packet )
               THEN
          }ACTION
}STATE

3 STATE{
     DL-REL#CONF ?L2_SERVICE
          ACTION{
               T." RELEASE IS CONFIRMED " TCR
               0 NEW_STATE
          }ACTION
}STATE
```

11.4. TEST3.F

```
(-------------------------------------------------------------------------)
( File      :TEST3.F                                                      )
(-------------------------------------------------------------------------)
( Function :To test the normal Data Link establishment, data transfer     )
(           and release; Peer is the NETWORK, SAPI is set to be 0,        )
(           TEI is assigned automatically; Peer only receives data.       )
(-------------------------------------------------------------------------)
( Note      :Testscript is for user emulation only.                       )


TCLR                                ( Clear state machine )
MONITOR                             ( Turn on monitor mode )

     " *** HELLO, THIS IS SIEMENS K1197 ***" MAKE_DATA1

0 STATE_INIT{
     " YOU ARE THE USER " MAG_BG DISPLAY
     ." Press F1 to establish the connection!" CR
     BLU_BG TCOLOR                  ( Set the color of trace lines )
     L2_ON                          ( Make sure layer 2 emulation is active )
}STATE_INIT

0 STATE{
     FK_F1
          ACTION{                   ( Detect function key 1 )
               DL_ESTABLISH
               T." Link establishment initiated" TCR
               1 NEW_STATE
          }ACTION

}STATE

1 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

1 STATE{
     DL-EST#CONF ?L2_SERVICE
          ACTION{
               2 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE_INIT{
     ." Press F1 to release, F2 to send data " CR
     " RELEASE  " 1 =TM_KEY
     " SEND DATA" 2 = TM_KEY
}STATE_INIT

2 STATE{
     FK_F1                          ( Detect function key 1 )
          ACTION{
               DL_RELEASE           ( Release the link )
               3 NEW_STATE
          }ACTION

     FK_F2                          ( Detect function key 2 )
          ACTION{
               1 PKT DL_DATA        ( Send the data packet type 1 )
          }ACTION
}STATE

3 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

3 STATE{
     DL-REL#CONF ?L2_SERVICE        ( Wait for release confirmation event )
          ACTION{
               T." *** TEST DONE *** " TCR
               0 NEW_STATE
          }ACTION
}STATE
```

11.5. TEST4.F

```
(-------------------------------------------------------------------------)
( File      : TEST4                                                       )
( Function  : To test the receiving capability of the peer NETWORK Prompt )
(             for the number of send frame, and the number of bytes in a frame )
(-------------------------------------------------------------------------)
( Note      : Testscript is for user emulation only.                      )

#IFDEF DISPLAY_NOCR
     FORGET DISPLAY_NOCR
#ENDIF

: DISPLAY_NOCR                ( ADDR/COLOR --- )
          >R COUNT R> STYPE_A
;

TCLR                                  ( Clear the state machine )
MONITOR                               ( Turn on monitor mode )

0 STATE_INIT{
     POP_COMMAND
     " YOU ARE THE USER " MAG_BG DISPLAY
     ." Press F1 to start " CR
     CLOSE_WINDOW
     " START     "  1 =TM_KEY
     BLU_BG TCOLOR                    ( set the color of trace lines )
}STATE_INIT

0 STATE{
     FK_F1
          ACTION{
               DL_ESTABLISH           ( Establish the data link )
               T." Link establishment initiated" TCR
               1 NEW_STATE
          }ACTION
}STATE

1 STATE_INIT{
     TM_DEF_KEYS                      ( Reset Function Key labels )
}STATE_INIT

1 STATE{
     DL-EST#CONF ?L2_SERVICE          ( Wait for an establish confirmation event)
               ACTION{
               " Press F1 to release, F2 to send data " BLK_BG DISPLAY
               2 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE_INIT{
     ." Press F1 to release, F2 to send data " CR
     " RELEASE  " 1 =TM_KEY
     "   SEND   " 2 = TM_KEY
}STATE_INIT

2 STATE{
     FK_F1
          ACTION{                    ( Detect function key 1 )
               DL_RELEASE            ( Release the link )
               5 NEW_STATE
          }ACTION

     FK_F2
          ACTION{                    ( Detect function key 2 )
               " Enter the number of packets to be sent : "
               BLU_BG DISPLAY_NOCR
               3 NEW_STATE
          }ACTION
}STATE

3 STATE_INIT{
     TM_DEF_KEYS                     ( Reset Function Key labels )
}STATE_INIT

3 STATE{
     ?KEYBOARD
          ACTION{                    ( Detect a keyboard entry )
               EXPECT_NUM            ( Is it a numeric entry )
               IF N-PKT* !           ( Store the valid entry )
                    " Enter the number of bytes in a packet : "
                    BLU_BG DISPLAY_NOCR
                    4 NEW_STATE
               ELSE
                    DROP             ( Throw away invalid entry )
                    " Invalid Entry " RED_BG DISPLAY
                    " Enter the number of bytes in a packet :"
                    RED_BG DISPLAY_NOCR
               THEN
          }ACTION
}STATE
```

```
4 STATE{
     ?KEYBOARD
          ACTION{                    ( Detect a keyboard entry )
               EXPECT_NUM            ( Is it a numeric entry )
               IF COUNT* !           ( Store the valid entry )
                    SEND             ( Send the packets )
                    2 NEW_STATE
               ELSE
                    DROP             ( Throw away invalid entry )
                    " Invalid Entry " RED_BG DISPLAY
                    " Enter the number of bytes in a packet :"
                    RED_BG DISPLAY_NOCR
               THEN
          }ACTION
}STATE

5 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

5 STATE{
     DL-REL#CONF ?L2_SERVICE        ( Wait for a release confirmation event )
          ACTION{
               MAG_BG TCOLOR T." TEST DONE " BLU_BG TCOLOR TCR
               0 NEW_STATE
          }ACTION
}STATE
```

11.6. TEST5.F

```
(-------------------------------------------------------------------------)
( File      : TEST5.F                                                     )
(-------------------------------------------------------------------------)
( Function : To test the peer NETWORK for receiving an out-of-sequence    )
(            I frame Peer is the NETWORK, SAPI is set to be 0,            )
(            TEI is assigned automatically.The local user will initiate   )
(            the data link automatically                                  )
(-------------------------------------------------------------------------)
( Note      : Testscript is for user emulation only.                      )

TCLR                                    ( Clear the state machine )
MONITOR                                 ( Turn on monitor mode )

     " *** HELLO, THIS IS SIEMENS K1197 ***" MAKE_DATA1

0 STATE_INIT{
          CR CR CR                      ( Clear Command Window )
          " YOU ARE THE USER " MAG_BG DISPLAY
          " Press F1 to start " BLK_BG DISPLAY
}STATE_INIT

0 STATE{
     FK_F1                              ( Detect function key 1 )
          ACTION{
               DL_ESTABLISH             ( Establish the link )
               1 NEW_STATE
          }ACTION
}STATE

1 STATE{
     DL-EST#CONF ?L2_SERVICE            ( Wait for an establish confirmation event)
          ACTION{
               2 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE_INIT{
     ." Press F1 to release, F2 to send data " BLU_BG_DISPLAY
     ."       F3 to send INVALID packet      " BLU_BG_DISPLAY
     " RELEASE  " 1 =TM_KEY
     " SEND-DATA" 2 =TM_KEY
     "  INV-N(S)" 3 =TM_KEY
}STATE_INIT

2 STATE{
     FK_F1
          ACTION{                    ( Detect function key 1 )
               DL_RELEASE            ( Release the link )
               4 NEW_STATE
          }ACTION
     FK_F2
          ACTION{                    ( Detect function key 2 )
               1 PKT SENDP           ( Send data packet message type 1 )
          }ACTION
     FK_F3
          ACTION{                    ( Detect function key 3 )
               2 K                   ( Window size must be bigger than 1,
                                         nothing will be sent )
               1 VS* +!              ( Set the invalid send number )
               1 PKT SENDP           ( Send the out-of-sequence frame )
               RED_BG TCOLOR
               T." An out-of-sequence INFO frame has been sent " TCR
               BLU_BG TCOLOR
               3 NEW_STATE
          }ACTION
}STATE

3 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

3 STATE{                            ( ERROR RECOVERY STATE )
     R#REJ ?RX_FRAME                ( Wait for REJ command frame )
          ACTION{
               MAG_BG TCOLOR
               T." REJ COMMAND frame has been received " TCR
               T." State machine will send a correct frame " TCR
               2 NEW_STATE
               1 K                  ( Set window size back to original value )
          }ACTION
}STATE
```

```
4 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

4 STATE{
     DL-REL#CONF ?L2_SERVICE
          ACTION{
               T." TEST DONE " TCR
          }ACTION
}STATE
```

11.7. TEST6.F

```
(-------------------------------------------------------------------------)
( File      :TEST6.F                                                      )
(-------------------------------------------------------------------------)
( Purpose   :To test the peer NETWORK for receiving an out-of-sequence    )
(            RR frame Peer is the NETWORK, SAPI is set to be 16,          )
(            TEI is assigned automatically.The local user will initiate   )
(            the data link automatically                                  )
(-------------------------------------------------------------------------)
( Note      :Testscript is for user emulation only.                       )


#IFNOTDEF COUNTER1
          0 VARIABLE COUNTER1
#ENDIF

TCLR                                ( Clear the state machine )
MONITOR                             ( Turn on monitor mode )

0 STATE_INIT{
          " YOU ARE THE USER " MAG_BG DISPLAY
          ." Press F1 to start " BLK_BG DISPLAY
          "  START   " 1 =TM_KEY    ( Make sure, layer 2 emulation is active )
}STATE_INIT

0 STATE{
     FK_F1                          ( Detect function key 1 )
          ACTION{
               DL_ESTABLISH         ( Establish the link )
               0 COUNTER1 !         ( Initialize the counter )
               1 NEW_STATE
          }ACTION
}STATE

1 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

1 STATE{
     DL-EST#CONF ?L2_SERVICE        ( Wait for an establish confirmation event)
     ACTION{
          MAG_BG TCOLOR T." Data Link has been established " BLU_BG TCOLOR TCR
          L2_OFF                    ( Turn layer 2 state machine off )
          2 NEW_STATE
     }ACTION
}STATE
```

```
2 STATE_INIT{
     CR CR
     ." Press F1 to release, the link " CR
     " RELEASE  " 1 =TM_KEY
     T." Waiting for an I frame ...." TCR
}STATE_INIT

2 STATE{
     R#I ?RX_FRAME                ( Wait for the incoming I frame )
          ACTION{
               COUNTER1 @ 2 =     ( Send an invalid RR after 2 successful
                                       trans. )
               IF
                    2 VR* @ + NR  ( Set the invalid receive number )
                    S_RR          ( Send the out-of-sequence RR frame )
                    RED_BG TCOLOR
                    T." An out-of-sequence RR frame has been sent " TCR
                    BLU_BG TCOLOR
                    0 COUNTER1 !  ( Reinitialize the counter )
                    3 NEW_STATE
               ELSE
                    L2_EXEC       ( Execute the layer 2 state machine )
                    1 COUNTER +!  ( Increment the counter )
                    COUNTER! @ 2 = IF
                         T." Next I frame will be acknowledged incorrectly "
                    ELSE
                         T." Waiting for another I frame ...."
                    THEN
                    TCR
               THEN
          }ACTION

     FK_F1                        ( Detect function key 1 )
          ACTION{
               L2_ON              ( Turn layer 2 state machine on )
               DL_RELEASE         ( Release the link )
               4 NEW_STATE
          }ACTION
}STATE
```

```
3 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

3 STATE{                            ( Error Recovery State )
     R#SABME ?RX_FRAME
     ACTION{
          RED_BG TCOLOR
          T." SABME received as re-establishment of the link "
          TCR
          L2_EXEC                   ( Execute the layer 2 state machine )
          L2_ON                     ( Turn layer 2 state machine on )
          BLU_BG TCOLOR
          2 NEW_STATE
          0 COUNTER1 !
          L2_OFF
     }ACTION
}STATE

4 STATE{
     DL-REL#CONF ?L2_SERVICE        ( Wait for a release confirmation event )
          ACTION{
               T." TEST DONE " TCR
               0 NEW_STATE
          }ACTION
}STATE
```

11.8. TEST7.F

```
(-------------------------------------------------------------------------)
( File      : TEST7.F                                                     )
(-------------------------------------------------------------------------)
( Purpose   : To test the peer NETWORK for receiving a RNR frame Peer is  )
(             the NETWORK, TEI is assigned automatically. The local user  )
(             will initiate the data link automatically                   )
(-------------------------------------------------------------------------)
( Note      : Testscript is for user emulation only.                      )

TCLR                                ( Clear the state machine )
MONITOR                             ( Turn on monitor mode )

0 STATE_INIT{
          " YOU ARE THE USER " MAG_BG DISPLAY
          " Press F1 to start " CR
          TM_DEF_KEYS
          "  START    " 1 =TM_KEY
          L2_ON                     ( Turn layer 2 state machine on )
          0 TCOLOR                  ( Reset COLOR or TRACE lines )
}STATE_INIT

0 STATE{
     FK_F1                          ( Detect function key 1 )
          ACTION{
               DL_ESTABLISH         ( Establish the link )
               1 NEW_STATE
          }ACTION
}STATE

1 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

1 STATE{
     DL-EST#CONF ?L2_SERVICE        ( Wait for an establish confirmation event)
          ACTION{
               2 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE_INIT{
     ." Press F1 to release, the link " CR
     " RELEASE  " 1 =TM_KEY
     " SEND_RNR " 2 =TM_KEY
}STATE_INIT

2 STATE{
     FK_F1                          ( Detect function key 1 )
          ACTION{
               DL_RELEASE           ( Release the link )
               5 NEW_STATE
          }ACTION

     FK_F2
          ACTION{
               DL_SET_BUSY              ( Make the user busy, will send RNR
                                            F=0)
               L2_OFF               ( Turn layer 2 state machine off )
               RED_BG TCOLOR
               T." An RNR F=0 frame has been sent " TCR
               BLU_BG TCOLOR
               3 NEW_STATE
          }ACTION
}STATE


3 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

3 STATE{                            ( Error Recovery State )
     R#RR ?RX_FRAME                 ( Wait for a RR response frame )
     ACTION{
     L2_EXEC                  ( Execute the layer 2 state machine )
          RED_BG TCOLOR
          T." RR frame has been received "TCR
          BLU_BG TCOLOR
          4 NEW_STATE
     }ACTION
}STATE


4 STATE{                            ( Error Recovery State )
     L2_EXEC                        ( Execute the layer 2 state machine )
     DL_NOT_BUSY                    ( Reset the user to be not busy )
                                    ( Will send a RR F=0 )
     L2_EXEC                        ( Execute the layer 2 state machine )
     L2_ON                          ( Turn layer 2 state machine on )
     RED_BG TCOLOR
     T." A RR F=0 frame has been sent, "TCR
     BLU_BG TCOLOR
     2 NEW_STATE
}STATE
```

```
5 STATE{
     DL-REL#CONF ?L2_SERVICE        ( Wait for a release confirmation event )
          ACTION{
               T." TEST DONE " TCR
               0 NEW_STATE
          }ACTION
}STATE
```

11.9. TEST8.F

```
(-------------------------------------------------------------------------)
( File      : TEST8.F                                                     )
(-------------------------------------------------------------------------)
( Purpose   : To test the peer NETWORK for receiving an out-of-sequence   )
(             RR frame. Peer is the NETWORK,TEI is assigned automatically.)
(             The local user will initiate the data link automatically    )
(-------------------------------------------------------------------------)
( Note      : Testscript is for user emulation only.                      )

TCLR                                    ( Clear the state machine )
MONITOR                                 ( Turn on monitor mode )

0 STATE_INIT{
          " YOU ARE THE USER " MAG_BG DISPLAY
          ." Press F1 to start " CR
          "  START  " 1 =TM_KEY
          L2_ON                         ( Turn layer 2 state machine on )
}STATE_INIT

0 STATE{
     FK_F1                              ( Detect function key 1 )
          ACTION{
               DL_ESTABLISH             ( Establish the link )
               1 NEW_STATE
          }ACTION
}STATE

1 STATE{
     DL-EST#CONF ?L2_SERVICE            ( Wait for an establish confirmation event)
          ACTION{
               MAG_BG TCOLOR
               T." Data Link has been established "
               BLU_BG TCOLOR TCR
               2 NEW_STATE
          }ACTION
}STATE
```

```
2 STATE_INIT{
     ." Press F1 to release, the link, F2 to send an out-of-seq RR " CR
     " RELEASE  " 1 =TM_KEY
     " BAD RR   " 2 =TM_KEY
}STATE_INIT

2 STATE{
     FK_F2                          ( Detect function key 2 )
          ACTION{
               VR* @ 1 - NR         ( Set the invalid receive number )
               S_RR                 ( Send the out-of-sequence RR frame )
               RED_BG TCOLOR
               T. " An out-of-sequence RR frame has been sent " TCR
               BLU_BG TCOLOR
               L2_OFF               ( Turn layer 2 state machine off )
               3 NEW_STATE
          }ACTION

     FK_F1                          ( Detect function key 1 )
          ACTION{
               DL_RELEASE           ( Release the link )
               4 NEW_STATE
          }ACTION
}STATE

3 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

3 STATE{                            ( Error Recovery State )
     R#SABME ?RX_FRAME                   ( Wait for a re-establishment )
     ACTION{
     RED_BG TCOLOR
     T." SABME received as re-establishment of the link " TCR
     BLU_BG TCOLOR
     L2_EXEC                        ( Execute the layer 2 state machine )
     L2_ON                          ( Turn layer 2 state machine on )
     1 NEW_STATE
     }ACTION
}STATE

4 STATE_INIT{
     TM_DEF_KEYS                    ( Reset Function Key labels )
}STATE_INIT

4 STATE{
     DL-REL#CONF ?L2_SERVICE        ( Wait for a release confirmation event )
          ACTION{
               T." TEST DONE " TCR
               0 NEW_STATE
          }ACTION
}STATE
```

INDEX

Symbols

?ABORT, 4-4
?COMMAND, 4-1
?CRC_ERROR, 4-4
?KEY, 4-7
?KEYBOARD, 4-8
?L1_EVENT, 4-5
?L2_SERVICE, 4-5
?L3_IE, 9-1, 9-3
?L3_IE_ERROR, 9-1, 9-5
?L3_IE_ORDER, 9-1, 9-3
?L3_IECOUNT, 9-1, 9-2
?L3_MAND_IE, 9-1, 9-3
?L3_MAND_IE_ERR, 9-1, 9-3
?L3_MAND_IE_MISSING, 9-1, 9-3
?L3_MAND_IE_ORDER, 9-1, 9-3
?L3_MATCH, 9-1, 9-2
?L3_MSG, 9-1
?L3_OCTET, 9-1, 9-4
?L3_OPT_IE, 9-1, 9-3
?L3_OPT_IE_ERR, 9-1, 9-3
?L3_OPT_IE_ORDER, 9-1, 9-3
?L3_SEARCH, 9-1, 9-2
?L3_UNEXP_IE, 9-1, 9-4
?L3_UNREC_IE, 9-1, 9-3
?L3_VALID_MSG, 9-1, 9-2
?PIGGYBACKED, 4-6
?RESPONSE, 4-1
?RX_FRAME, 4-2
?RX_MATCH, 4-3
?RX_SEARCH, 4-3
?TEST_EVENT, 4-10
?TIMER, 4-4
?TM, 2-3
?TM_IPC, 4-7
?WAKEUP, 4-1
}ACTION, 3-1
}SEQ, 3-2
}STATE, 3-1
}STATE_INIT, 3-2
$MSG_CR_FLAG, 8-1
$MSG_CR_LEN, 8-1
$MSG_CR_VALUE, 8-1
$MSG_DISCR, 8-1
$MSG_ERROR, 8-1
$MSG_ID, 8-1
$MSG_TYPE, 8-1
=TM, 2-3
=TM_KEY, 5-2

A

ACTION{, 3-1

B

BLK_BG, 5-1
BLU_BG, 5-1
BUFFER, 5-4
BUFFER_LENGTH, 5-4

C

CTRACE, 6-1
CYA_FG, 5-1

D

DISPLAY, 5-1
DTRACE, 6-1

E

EXECF, 2-2
EXECTS, 2-2
EXP-EXTEND#, 9-5

F

FILE->BUFFER, 5-4
FK_F1, 4-8
FK_F2, 4-8
FK_F3, 4-8
FK_F4, 4-8
FK_F5, 4-8
FK_F6, 4-8

G

GET_IES, 9-1, 9-4
GLOB_TCLR, 2-3
GLOB_TM_RUN, 2-3
GLOB_TM_STOP, 2-3
GRN_FG, 5-1

I

IE-TOO-LONG#, 9-5
IE-TOO-SHORT#, 9-5
INV-CALLREF#, 8-1
INV-DISCR#, 8-1

L

LOAD_RETURN_STATE, 3-2

M

MAG_BG, 5-1
MENU_OFF, 4-7
MENU_ON, 4-7
MILLISECONDS_ELAPSED, 5-2
MINUTES_ELAPSED, 5-2

N

NEW_STATE, 3-2
NEW_TM, 2-2
NO-ERROR#, 8-1
NO-IE-ERROR#, 9-5
NO-MSG#, 8-1

O

OTHER_EVENT, 4-9

R

RANGE-ERR#, 8-1
RED_BG, 5-1
RETURN_STATE, 3-2
REV_VIDEO, 5-1
RTRACE, 6-1
RUN_SEQ, 3-2

S

SAVETS, 2-2
SECONDES_ELAPSED, 5-2
SEQ{, 3-2
SET_EVENT_FLAG, 4-10
SHIFT-ERR#, 8-1
SHORT-MSG#, 8-1
START_LAPSE_TIMER, 5-2
START_TM_TIMER, 5-3
STATE{, 3-1
STATE-TRACE, 6-1
STATE_INIT{, 3-2
STOP_ALL_TIMERS, 5-3
STOP_TM_TIMER, 5-3
STRING->BUFFER, 5-4

T

T., 6-1
T.", 6-1
T.H, 6-1
T.TYPE, 6-1
TCLR, 2-2
TCOLOR, 6-1
TCR, 6-1
TEMIT, 6-1
TM.RAM, 2-3
TM_IPC_SEND, 5-3
TM_RUN, 2-2
TM_STOP, 2-2
TM0, 2-3
TM1, 2-3
TM2, 2-3
TM3, 2-3
TM4, 2-3
TM5, 2-3
TM6, 2-3
TM7, 2-3

U

UNEXP-EXTEND#, 9-5

W

WAKEUP_OFF, 2-3
WAKEUP_ON, 2-3
WHI_FG, 5-1

Y

YEL_FG, 5-1

INDEX

Symbols

?ABORT: *TEST MANAGER* 4-4

?B/D_CALLED: *SIMULATION* 6-19

?B/D_CALLING: *SIMULATION* 6-19

?B/D_CONN_CNF: *SIMULATION* 6-18

?B/D_CONN_IND: *SIMULATION* 6-18

?B/D_DET_IND: *SIMULATION* 6-18

?B/D_DISC_CNF: *SIMULATION* 6-18

?B/D_DISC_IND: *SIMULATION* 6-18

?B/D_UPTR: *SIMULATION* 6-19

?BC_ALLOC: *SIMULATION* 6-12, 6-26

?BC_CONN: *SIMULATION* 6-12, 6-26

?BC_SELECT: *SIMULATION* 6-12, 6-26

?CN: *SIMULATION* 6-8, 6-22

?CN_CALLED_NUM: *SIMULATION* 6-9, 6-23

?CN_CALLED_SAD: *SIMULATION* 6-9, 6-23

?CN_CALLING_NUM: *SIMULATION* 6-9, 6-23

?CN_CALLING_SAD: *SIMULATION* 6-9, 6-23

?CN_CEI: *SIMULATION* 6-9, 6-24

?CN_CHANNEL_NUM: *SIMULATION* 6-9, 6-23

?CN_CHANNEL_TYPE: *SIMULATION* 6-9, 6-23

?CN_CR_SOURCE: *SIMULATION* 6-10, 6-24

?CN_CR_VALUE: *SIMULATION* 6-24

?CN_IE_BUFFER: *SIMULATION* 6-11, 6-25

?CN_IE_LIST: *SIMULATION* 6-25

?CN_PD: *SIMULATION* 6-9, 6-23

?CN_TIMER: *SIMULATION* 6-13, 6-20

?CN_TIMER_DUR: *SIMULATION* 6-13

?COMMAND: *TEST MANAGER* 4-1

?CR_LENGTH: *SIMULATION* 6-10, 6-24

?CRC_ERROR: *TEST MANAGER* 4-4

?KEY: *TEST MANAGER* 4-7

?KEYBOARD: *TEST MANAGER* 4-8

?L1_EVENT: *TEST MANAGER* 4-5

?L2_ON: *SIMULATION* 4-14

?L2_SERVICE: *TEST MANAGER* 4-5

?L3_ID_COUNT: *SIMULATION* 6-20

?L3_IE: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-3

?L3_IE_ERROR: *TEST MANAGER* 9-1, 9-5

?L3_IE_ORDER: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-3

?L3_IECOUNT: *TEST MANAGER* 9-1, 9-2

?L3_MAND_IE: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-3

?L3_MAND_IE_ERR: *TEST MANAGER* 9-1, 9-3

?L3_MAND_IE_MISSING: *TEST MANAGER* 9-1, 9-3

?L3_MAND_IE_ORDER: *TEST MANAGER* 9-1, 9-3

?L3_MATCH: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-2

?L3_MSG: *SIMULATION* 6-20; *TEST MANAGER* 9-1

?L3_OCTET: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-4

?L3_OPT_IE: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-3

?L3_OPT_IE_ERR: *TEST MANAGER* 9-1, 9-3

?L3_OPT_IE_ORDER: *TEST MANAGER* 9-1, 9-3

?L3_SEARCH: *TEST MANAGER* 9-1, 9-2

?L3_UNEXP_IE: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-4

?L3_UNREC_IE: *TEST MANAGER* 9-1, 9-3

?L3_VALID_MSG: *SIMULATION* 6-20; *TEST MANAGER* 9-1, 9-2

?MESSAGE: *SIMULATION* 6-28

?MESSAGE_COMMENT: *SIMULATION* 6-28

?MESSAGE_DATA: *SIMULATION* 6-28

?MESSAGE_FREE: *SIMULATION* 6-29

?MESSAGE_NAME: *SIMULATION* 6-28

?MESSAGE_NUMBER: *SIMULATION* 6-28

?ML_ON: *SIMULATION* 4-15

?PIGGYBACKED: *TEST MANAGER* 4-6

?POOL_CR_FLAG: *SIMULATION* 6-30

?POOL_CR_VALUE: *SIMULATION* 6-30

?PTP: *SIMULATION* 4-14

?RESPONSE: *TEST MANAGER* 4-1

?RX_FRAME: *TEST MANAGER* 4-2

?RX_MATCH: *TEST MANAGER* 4-3

?RX_SEARCH: *TEST MANAGER* 4-3

?TEST_EVENT: *TEST MANAGER* 4-10

?TIMER: *TEST MANAGER* 4-4
?TM: *TEST MANAGER* 2-3
?TM_IPC: *TEST MANAGER* 4-7
?WAKEUP: *TEST MANAGER* 4-1
}ACTION: *TEST MANAGER* 3-1
}SEQ: *TEST MANAGER* 3-2
}STATE: *TEST MANAGER* 3-1
}STATE_INIT: *TEST MANAGER* 3-2
$MSG_CR_FLAG: *TEST MANAGER* 8-1
$MSG_CR_LEN: *TEST MANAGER* 8-1
$MSG_CR_VALUE: *TEST MANAGER* 8-1
$MSG_DISCR: *TEST MANAGER* 8-1
$MSG_ERROR: *TEST MANAGER* 8-1
$MSG_ID: *TEST MANAGER* 8-1
$MSG_TYPE: *TEST MANAGER* 8-1
+ACTIVATE: *MONITOR* 9-2
+BPVL: *MONITOR* 9-3
+CAPT: *MONITOR* 9-5
+CEPTSPEC: *MONITOR* 9-3
+CRCR: *MONITOR* 9-3
+DEACTICATE: *MONITOR* 9-2
+DISC: *MONITOR* 9-3
+DISK: *MONITOR* 9-5
+DLCI: *MONITOR* 9-5
+DM: *MONITOR* 9-3
+FERR: *MONITOR* 9-3
+FRMR: *MONITOR* 9-3
+I: *MONITOR* 9-3
+INVFRM: *MONITOR* 9-3
+LST_PLOCK: *MONITOR* 9-3
+LST_SIGL: *MONITOR* 9-3
+LSTFRM: *MONITOR* 9-2
+MERR: *MONITOR* 9-3
+NORMAL: *MONITOR* 9-3
+RECOVERY: *MONITOR* 9-2
+RED_ALARM: *MONITOR* 9-3
+REJ: *MONITOR* 9-3
+RI2ERR: *MONITOR* 9-2
+RNR: *MONITOR* 9-3
+RR: *MONITOR* 9-3
+SABME: *MONITOR* 9-3
+SAPI: *MONITOR* 9-5
+STRING: *MONITOR* 9-4
+TEI: *MONITOR* 9-5
+TIME: *MONITOR* 9-5
+UA: *MONITOR* 9-3
+UI: *MONITOR* 9-3
+UNDEFINED: *MONITOR* 9-2
+XID: *MONITOR* 9-3
+YEL_ALARM: *MONITOR* 9-3
-ACTIVATE: *MONITOR* 9-2
-BPVL: *MONITOR* 9-3
-CAPT: *MONITOR* 9-5
-CEPTSPEC: *MONITOR* 9-3
-CRCR: *MONITOR* 9-3
-DEACTIVATE: *MONITOR* 9-2
-DISC: *MONITOR* 9-3
-DISK: *MONITOR* 9-5
-DLCI: *MONITOR* 9-5
-DM: *MONITOR* 9-3
-F_C_REF: *MONITOR* 8-8, 8-12
-F_P_DISC: *MONITOR* 8-4, 8-8, 8-12
-FERR: *MONITOR* 9-3
-FRMR: *MONITOR* 9-3
-I: *MONITOR* 9-3
-INVFRM: *MONITOR* 9-3
-LST_PLOCK: *MONITOR* 9-3
-LST_SIGL: *MONITOR* 9-3
-LSTFRM: *MONITOR* 9-2
-MERR: *MONITOR* 9-3
-NORMAL: *MONITOR* 9-3
-RECOVERY: *MONITOR* 9-2
-RED_ALARM: *MONITOR* 9-3
-REJ: *MONITOR* 9-3
-RI2ERR: *MONITOR* 9-2
-RNR: *MONITOR* 9-3
-RR: *MONITOR* 9-3
-SABME: *MONITOR* 9-3
-SAPI: *MONITOR* 9-5
-STRING: *MONITOR* 9-4
-T_C_REF: *MONITOR* 9-5
-T_P_DISC: *MONITOR* 9-5
-TEI: *MONITOR* 9-5
-TIME: *MONITOR* 9-5
-UA: *MONITOR* 9-3

-UI: *MONITOR* 9-3
-UNDEFINED: *MONITOR* 9-2
-XID: *MONITOR* 9-3
-YEL-ALARM: *MONITOR* 9-3
/CN_CR_VALUE: *SIMULATION* 6-10
=ALL: *MONITOR* 9-4
=B/D_CALLED: *SIMULATION* 6-19
=B/D_CALLING: *SIMULATION* 6-19
=B/D_UPTR: *SIMULATION* 6-19
=BOTH: *MONITOR* 9-2
=CF_SAPI: *MONITOR* 8-8
=CF_TEI: *MONITOR* 8-7
=CN: *SIMULATION* 6-8, 6-22
=CN_CALLED_NUM: *SIMULATION* 6-9, 6-23
=CN_CALLED_SAD: *SIMULATION* 6-9, 6-23
=CN_CALLING_NUM: *SIMULATION* 6-9, 6-23
=CN_CALLING_SAD: *SIMULATION* 6-9, 6-23
=CN_CEI: *SIMULATION* 6-9, 6-23
=CN_CHANNEL_NUM: *SIMULATION* 6-9, 6-23
=CN_CHANNEL_TYPE: *SIMULATION* 6-9, 6-23
=CN_CR_SOURCE: *SIMULATION* 6-10, 6-24
=CN_CR_VALUE: *SIMULATION* 6-10, 6-24
=CN_PD: *SIMULATION* 6-9, 6-23
=CN_TIMER_DUR: *SIMULATION* 6-13
=CR_LENGTH: *SIMULATION* 6-10, 6-24
=DF_SAPI: *MONITOR* 8-12
=DF_TEI: *MONITOR* 8-11
=F_C_REF: *MONITOR* 8-4, 8-8, 8-12
=F_P_DISC: *MONITOR* 8-4, 8-8, 8-12
=ICHAR: *SIMULATION* 4-16
=NAT_BITS: *SIMULATION* 4-16
=NET: *MONITOR* 9-2
=NONE: *MONITOR* 9-4
=PLAYBACK: *MONITOR* 9-2
=POOL_CR_FLAG: *SIMULATION* 6-30
=POOL_CR_VALUE: *SIMULATION* 6-30
=RF_SAPI: *MONITOR* 8-4
=RF_TEI: *MONITOR* 8-3
=SAPI: *MONITOR* 9-5
=STRING: *MONITOR* 9-4
=T_C_REF: *MONITOR* 9-5
=T_P_DISC: *MONITOR* 9-5
=TEI: *MONITOR* 9-5
=TIME: *MONITOR* 9-4
=TITLE: *MONITOR* 6-3
=TM: *TEST MANAGER* 2-3
=TM_KEY: *TEST MANAGER* 5-2
=TSLOT: *MONITOR* 7-3
=USR: *MONITOR* 9-2

A

ACTION{: *TEST MANAGER* 3-1
AI[]: *SIMULATION* 4-22
ALL_EXCLUDED: *SIMULATION* 6-35
ALL_INCLUDED: *SIMULATION* 6-35
ALLOC_MESSAGE: *SIMULATION* 6-29
ALTER_CALL_REF: *SIMULATION* 6-30
APPEND_IE: *SIMULATION* 6-31
ASPST: *SIMULATION* 4-8
ATEI: *SIMULATION* 4-12
AUTO_BROADCAST: *SIMULATION* 6-21
AUTO_HDR_UPDATE: *SIMULATION* 6-21
AUTO_MODE: *SIMULATION* 6-34
AUTO_UI_SETUP: *SIMULATION* 6-21

B

B/D_CONN_REQ: *SIMULATION* 6-18
B/D_CONN_RES: *SIMULATION* 6-18
B/D_DISC_REQ: *SIMULATION* 6-18
B/D_DISC_RES: *SIMULATION* 6-18
BACKWARD: *MONITOR* 7-2
BALM_OFF: *SIMULATION* 4-16
BALM_ON: *SIMULATION* 4-16
BC_ALLOC: *SIMULATION* 6-12, 6-26
BC_CONN: *SIMULATION* 6-12, 6-26
BC_DISC: *SIMULATION* 6-12, 6-26
BC_FREE: *SIMULATION* 6-12, 6-26
BC_INIT: *SIMULATION* 6-12, 6-26
BER: *SIMULATION* 4-12
BLK_BG: *TEST MANAGER* 5-1

BLOCK-COUNT: *SIMULATION* 4-20
BLOCK_ON: *MONITOR* 5-10
BLU_BG: *TEST MANAGER* 5-1
BOTTOM: *MONITOR* 7-2
BUFFER: *TEST MANAGER* 5-4
BUFFER_LENGTH: *TEST MANAGER* 5-4
BYTES: *SIMULATION* 4-10

C

C/I_OFF: *MONITOR* 5-9
C/I_ON: *MONITOR* 5-9
C1+ACTIVATED: *MONITOR* 8-5
C1+BPVL: *MONITOR* 8-6
C1+CEPTSPEC: *MONITOR* 8-6
C1+CRCR: *MONITOR* 8-6
C1+DEACTIVATED: *MONITOR* 8-5
C1+FERR: *MONITOR* 8-6
C1+LST_PLOCK: *MONITOR* 8-6
C1+LST_SIGL: *MONITOR* 8-6
C1+LSTFRM: *MONITOR* 8-5
C1+MERR: *MONITOR* 8-6
C1+NORMAL: *MONITOR* 8-6
C1+RECOVERY: *MONITOR* 8-5
C1+RED_ALARM: *MONITOR* 8-6
C1+RI2ERR: *MONITOR* 8-5
C1+UNDEFINED: *MONITOR* 8-5
C1+YEL_ALARM: *MONITOR* 8-6
C1-ACTIVATED: *MONITOR* 8-5
C1-BPVL: *MONITOR* 8-6
C1-CEPTSPEC: *MONITOR* 8-6
C1-CRCR: *MONITOR* 8-6
C1-DEACITVATED: *MONITOR* 8-5
C1-FERR: *MONITOR* 8-6
C1-LST_PLOCK: *MONITOR* 8-6
C1-LST_SIGL: *MONITOR* 8-6
C1-LSTFRM: *MONITOR* 8-5
C1-MERR: *MONITOR* 8-6
C1-NORMAL: *MONITOR* 8-6
C1-RECOVERY: *MONITOR* 8-5
C1-RED_ALARM: *MONITOR* 8-6
C1-RI2ERR: *MONITOR* 8-5
C1-YEL-ALARM: *MONITOR* 8-6
C1=ALL: *MONITOR* 8-5, 8-6
C1=NONE: *MONITOR* 8-5, 8-6
C1_UNDEFINED: *MONITOR* 8-5
C2+DISC: *MONITOR* 8-7
C2+DM: *MONITOR* 8-7
C2+FRMR: *MONITOR* 8-7
C2+I: *MONITOR* 8-7
C2+INV: *MONITOR* 8-7
C2+REJ: *MONITOR* 8-7
C2+RNR: *MONITOR* 8-7
C2+RR: *MONITOR* 8-7
C2+SABME: *MONITOR* 8-7
C2+UA: *MONITOR* 8-7
C2+UI: *MONITOR* 8-7
C2+XID: *MONITOR* 8-7
C2-DISC: *MONITOR* 8-7
C2-DM: *MONITOR* 8-7
C2-FRMR: *MONITOR* 8-7
C2-I: *MONITOR* 8-7
C2-INV: *MONITOR* 8-7
C2-REJ: *MONITOR* 8-7
C2-RNR: *MONITOR* 8-7
C2-RR: *MONITOR* 8-7
C2-SABME: *MONITOR* 8-7
C2-UA: *MONITOR* 8-7
C2-UI: *MONITOR* 8-7
C2-XID: *MONITOR* 8-7
C2=ALL: *MONITOR* 8-7
C2=NONE: *MONITOR* 8-7
CAPT_FULL: *MONITOR* 7-1
CAPT_OFF: *MONITOR* 7-1
CAPT_ON: *MONITOR* 7-1
CAPT_WRAP: *MONITOR* 7-1
CES: *SIMULATION* 4-12
CES=0: *SIMULATION* 4-12
CES=1..7: *SIMULATION* 4-12
CF_SAPI: *MONITOR* 8-8
CF_SAPI=ALL: *MONITOR* 8-8
CF_TEI: *MONITOR* 8-7
CF_TEI=127: *MONITOR* 8-7
CF_TEI=ALL: *MONITOR* 8-7
CF_TEI127: *MONITOR* 8-7

CLEAR_CR: *SIMULATION* 6-10, 6-24
CLEAR_CRT: *MONITOR* 7-1
CLEAR_MESSAGE: *SIMULATION* 6-29
CLEAR_POOL: *SIMULATION* 6-29
CLR POOL: *SIMULATION* 3-16
CLREC: *SIMULATION* 4-12
CMD/RESP*: *SIMULATION* 4-21
CN_INIT: *SIMULATION* 6-10, 6-15, 6-24
CN_START_TIMER: *SIMULATION* 6-13
CN_STOP_TIMER: *SIMULATION* 6-13
CN0: *SIMULATION* 6-8
CN0 ... CN7: *SIMULATION* 6-22
CN1: *SIMULATION* 6-8
CN2: *SIMULATION* 6-8
CN3: *SIMULATION* 6-8
CN4: *SIMULATION* 6-8
CN5: *SIMULATION* 6-8
CN6: *SIMULATION* 6-8
CN7: *SIMULATION* 6-8
COMMAND: *MONITOR* 4-2
COMPACT_POOL: *SIMULATION* 6-29
CONCAT: *SIMULATION* 3-16
CONCAT_POOLS: *SIMULATION* 6-29
COPY_BUFFERS: *SIMULATION* 6-11, 6-25
COPY_CN: *SIMULATION* 6-27
COPY_IE: *SIMULATION* 6-11, 6-25
COPY_IE_BUF: *SIMULATION* 6-11, 6-25
COPY_MESSAGE: *SIMULATION* 6-30
COPY_PARAMS: *SIMULATION* 6-11, 6-25
CR_TO_CN: *SIMULATION* 6-27
CR_VAL_MAX: *SIMULATION* 6-10, 6-24
CRC_ERROR: *SIMULATION* 4-11
CS=ASCII: *MONITOR* 5-10
CS=EBCDIC: *MONITOR* 5-10
CS=HEX: *MONITOR* 5-10
CTOD_OFF: *MONITOR* 7-2
CTOD_ON: *MONITOR* 7-2
CTRACE: *TEST MANAGER* 6-1
CYA_FG: *TEST MANAGER* 5-1

D

D1+ACTIVATED: *MONITOR* 8-9
D1+BPVL: *MONITOR* 8-10
D1+CEPTSPEC: *MONITOR* 8-10
D1+CRCR: *MONITOR* 8-10
D1+DEACTIVATED: *MONITOR* 8-9
D1+FERR: *MONITOR* 8-10
D1+LST_PLOCK: *MONITOR* 8-10
D1+LST_SIGL: *MONITOR* 8-10
D1+LSTFRM: *MONITOR* 8-9
D1+MERR: *MONITOR* 8-10
D1+NORMAL: *MONITOR* 8-10
D1+RECOVERY: *MONITOR* 8-9
D1+RED_ALARM: *MONITOR* 8-10
D1+RI2ERR: *MONITOR* 8-9
D1+UNDEFINED: *MONITOR* 8-9
D1+YEL_ALARM: *MONITOR* 8-10
D1-ACTIVATED: *MONITOR* 8-9
D1-BPVL: *MONITOR* 8-10
D1-CEPTSPEC: *MONITOR* 8-10
D1-CRCR: *MONITOR* 8-10
D1-DEACTIVATED: *MONITOR* 8-9
D1-FERR: *MONITOR* 8-10
D1-LST_PLOCK: *MONITOR* 8-10
D1-LST_SIGL: *MONITOR* 8-10
D1-LSTFRM: *MONITOR* 8-9
D1-MERR: *MONITOR* 8-10
D1-NORMAL: *MONITOR* 8-10
D1-RECOVERY: *MONITOR* 8-9
D1-RED_ALARM: *MONITOR* 8-10
D1-RI2ERR: *MONITOR* 8-9
D1-UNDEFINED: *MONITOR* 8-9
D1-YEL_ALARM: *MONITOR* 8-10
D1=ALL: *MONITOR* 8-9, 8-10
D1=NONE: *MONITOR* 8-9, 8-10
D2+DISC: *MONITOR* 8-11
D2+DM: *MONITOR* 8-11
D2+FRMR: *MONITOR* 8-11
D2+I: *MONITOR* 8-11
D2+INV: *MONITOR* 8-11
D2+REJ: *MONITOR* 8-11
D2+RNR: *MONITOR* 8-11

D2+RR: *MONITOR* 8-11
D2+SABME: *MONITOR* 8-11
D2+UA: *MONITOR* 8-11
D2+XID: *MONITOR* 8-11
D2-DISC: *MONITOR* 8-11
D2-DM: *MONITOR* 8-11
D2-FRMR: *MONITOR* 8-11
D2-I: *MONITOR* 8-11
D2-INV: *MONITOR* 8-11
D2-REJ: *MONITOR* 8-11
D2-RNR: *MONITOR* 8-11
D2-RR: *MONITOR* 8-11
D2-SABME: *MONITOR* 8-11
D2-UA: *MONITOR* 8-11
D2-UI: *MONITOR* 8-11
D2-XID: *MONITOR* 8-11
D2=ALL: *MONITOR* 8-11
D2=NONE: *MONITOR* 8-11
D2_UI: *MONITOR* 8-11
DATA-STATUS: *SIMULATION* 4-23
DELETE_IE: *SIMULATION* 6-31
DF_DAPI=ALL: *MONITOR* 8-12
DF_SAPI: *MONITOR* 8-12
DF_TEI: *MONITOR* 8-11
DF_TEI=127: *MONITOR* 8-11
DF_TEI=ALL: *MONITOR* 8-11
DF_TEI127: *MONITOR* 8-11
DIS_REC: *MONITOR* 6-3
DISC: *SIMULATION* 4-1
DISK_OFF: *MONITOR* 6-3
DISPLAY: *TEST MANAGER* 5-1
DL_CLR_BUSY: *SIMULATION* 4-5
DL_DATA: *SIMULATION* 4-5
DL_DATA_REQ: *SIMULATION* 4-5, 6-4
DL_ESTABLISH: *SIMULATION* 4-5, 6-4
DL_RELEASE: *SIMULATION* 4-5, 6-4
DL_SET_BUSY: *SIMULATION* 4-5
DL_UNIT_DATA: *SIMULATION* 4-5
DL_UNIT_DATA_REQ: *SIMULATION* 4-5, 6-4
DM: *SIMULATION* 4-1
DO_ABORT: *SIMULATION* 4-11
DR0: *MONITOR* 7-1
DR1: *MONITOR* 7-1
DTRACE: *TEST MANAGER* 6-1

E

EDIT_MSG: *SIMULATION* 6-31
ENB_REC: *MONITOR* 6-3
END-TIME: *SIMULATION* 4-20
EVENT-TYPE: *SIMULATION* 4-20
EXCLUDED: *SIMULATION* 6-35
EXECF: *TEST MANAGER* 2-2
EXECTS: *TEST MANAGER* 2-2
EXP-EXTEND#: *TEST MANAGER* 9-5
EXT_BCH_OFF: *MONITOR* 7-3
EXT_BCH_ON: *MONITOR* 7-3
EXTERN_CLK: *SIMULATION* 4-17, 4-18

F

F0: *SIMULATION* 4-9
F1: *SIMULATION* 4-9
FAST_MODE: *MONITOR* 6-3
FER: *SIMULATION* 4-12
FERR=EVEN: *SIMULATION* 4-16
FERR=ODD: *SIMULATION* 4-16
FFMT=CRC4: *MONITOR* 7-3
FFMT=D4: *MONITOR* 7-3
FFMT=ESF: *MONITOR* 7-3
FFMT=PCM: *MONITOR* 7-3
FILE->BUFFER: *TEST MANAGER* 5-4
FK_F1: *TEST MANAGER* 4-8
FK_F2: *TEST MANAGER* 4-8
FK_F3: *TEST MANAGER* 4-8
FK_F4: *TEST MANAGER* 4-8
FK_F5: *TEST MANAGER* 4-8
FK_F6: *TEST MANAGER* 4-8
FORWARD: *MONITOR* 7-2
FRAME-ID: *SIMULATION* 4-22
FREEZE: *MONITOR* 4-2, 7-1
FRM_CHAR: *MONITOR* 5-9
FRM_COMP: *MONITOR* 5-9
FRM_HEX: *MONITOR* 5-9
FRM_OFF: *MONITOR* 5-9
FRM_SHRT: *MONITOR* 5-9

FRMR-C/R-BIT*: *SIMULATION* 4-21
FRMR-CNTL1*: *SIMULATION* 4-21
FRMR-CNTL2*: *SIMULATION* 4-21
FRMR-VR*: *SIMULATION* 4-21
FRMR-VS*: *SIMULATION* 4-21
FRMRW*: *SIMULATION* 4-21
FRMRX*: *SIMULATION* 4-21
FRMRY*: *SIMULATION* 4-21
FRMRZ*: *SIMULATION* 4-21
FROM_CAPT: *MONITOR* 6-3
FROM_DISK: *MONITOR* 6-3

G

GET_IES: *TEST MANAGER* 9-1, 9-4
GET_RX_MSG: *SIMULATION* 6-31
GLOB_TCLR: *TEST MANAGER* 2-3
GLOB_TM_RUN: *TEST MANAGER* 2-3
GLOB_TM_STOP: *TEST MANAGER* 2-3
GOOD_CRC: *SIMULATION* 4-11
GRN_FG: *TEST MANAGER* 5-1

H

HALT: *MONITOR* 4-2, 7-1
HELP: *MONITOR* 4-2
HIGHLIGHT: *MONITOR* 9-6

I

I/F=V.11: *MONITOR* 7-4
I/F=V.28: *MONITOR* 7-4
I/F=V.35: *MONITOR* 7-4
I/F=V.36: *MONITOR* 7-4
IDENTITY_ASSIGNED: *SIMULATION* 4-2
IDENTITY_CHECK_REQ: *SIMULATION* 4-2
IDENTITY_CHECK_RESP: *SIMULATION* 4-2
IDENTITY_DENIED: *SIMULATION* 4-2
IDENTITY_REMOVE: *SIMULATION* 4-2
IDENTITY_REQUEST: *SIMULATION* 4-2
IDENTITY_VERIFY: *SIMULATION* 4-2
IE-TOO-LONG#: *TEST MANAGER* 9-5
IE-TOO-SHORT#: *TEST MANAGER* 9-5
IHDLC_OFF: *MONITOR* 7-3
IHDLC_ON: *MONITOR* 7-3
IMPED=100: *SIMULATION* 4-17
IMPED=120: *MONITOR* 7-3
IMPED=75: *MONITOR* 7-3
IMPED=HI: *MONITOR* 7-3
IMPED=HIGH: *SIMULATION* 4-17
INCLUDED: *SIMULATION* 6-35
INIT_DEFAULT_TIMERS: *SIMULATION* 6-13
INIT_IE_BUFFERS: *SIMULATION* 6-11, 6-25
INSERT_ENTRY: *SIMULATION* 6-30
INT_BIT=0: *SIMULATION* 4-16
INT_BIT=1: *SIMULATION* 4-16
INTERN_CLK: *SIMULATION* 4-17
INV-CALLREF#: *TEST MANAGER* 8-1
INV-DISCR#: *TEST MANAGER* 8-1

K

K: *SIMULATION* 4-8
K*: *SIMULATION* 4-19

L

L1_COMP: *MONITOR* 5-9
L1_OFF: *MONITOR* 5-9
L1_SHRT: *MONITOR* 5-9
L2-LENGTH: *SIMULATION* 4-20
L2-POINTER: *SIMULATION* 4-20
L2_EXEC: *SIMULATION* 4-14
L2_OFF: *SIMULATION* 4-14
L2_ON: *SIMULATION* 4-14
L2_STATE_TRC: *SIMULATION* 4-15
L2_TIMER_TRC: *SIMULATION* 4-15
L2ST: *SIMULATION* 4-8
L3-LENGTH: *SIMULATION* 4-20
L3-POINTER: *SIMULATION* 4-20
L3_MUX_OFF: *SIMULATION* 6-14, 6-27
L3_MUX_ON: *SIMULATION* 6-14, 6-27
LCOD=AMI: *MONITOR* 7-3
LCOD=B8ZS: *MONITOR* 7-3
LCOD=HDB3: *MONITOR* 7-3
LENGTH=0: *MONITOR* 7-3
LENGTH=2: *MONITOR* 7-3

LOAD POOL: *SIMULATION* 3-16
LOAD_IE_BUFFER: *SIMULATION* 6-36
LOAD_MESSAGES: *SIMULATION* 6-29
LOAD_RETURN_STATE: *TEST MANAGER* 3-2
LONG_BUFFER: *SIMULATION* 6-11, 6-25

M

MADDR1*: *SIMULATION* 4-20
MADDR2*: *SIMULATION* 4-20
MAG_BG: *TEST MANAGER* 5-1
MAKE_DATA1: *SIMULATION* 4-13
MAKE_DATA1 / ... / MAKE_DATA8: *SIMULATION* 4-10
MAKE_DATA2/.../MAKE_DATA8: *SIMULATION* 4-13
MAN_MODE: *SIMULATION* 6-34
MCONTROL1*: *SIMULATION* 4-21
MCONTROL2*: *SIMULATION* 4-21
MDL_ALLOCATE: *SIMULATION* 4-6, 6-4
MDL_ASSIGN_I: *SIMULATION* 4-6
MDL_ASSIGN_R: *SIMULATION* 4-6
MDL_CURRENT: *SIMULATION* 4-6, 6-4
MDL_DEALLOCATE: *SIMULATION* 4-6, 6-4
MDL_ERROR_TRC: *SIMULATION* 4-15
MDL_EXISTED: *SIMULATION* 4-6, 6-4
MDL_INUSE: *SIMULATION* 6-4
MDL_REMOVE: *SIMULATION* 4-6, 6-4
MDL_SELECT: *SIMULATION* 4-6, 6-4
MDL_UNIT_DATA: *SIMULATION* 4-5
MENU: *MONITOR* 4-2
MENU_OFF: *TEST MANAGER* 4-7
MENU_ON: *TEST MANAGER* 4-7
MILLISECONDS_ELAPSED: *TEST MANAGER* 5-2
MINUTES_ELAPSED: *TEST MANAGER* 5-2
ML_EXEC: *SIMULATION* 4-15
ML_OFF: *SIMULATION* 4-15
ML_ON: *SIMULATION* 4-14
MLST: *SIMULATION* 4-8
MNR*: *SIMULATION* 4-21
MNS*: *SIMULATION* 4-21
MON_KEYS: *MONITOR* 4-2
MONITOR: *MONITOR* 4-2, 7-1
MOVE_MESSAGE: *SIMULATION* 6-30
MP/F-BIT*: *SIMULATION* 4-21
MSAPI*: *SIMULATION* 4-21
MSG_CHAR: *MONITOR* 5-9
MSG_COMP: *MONITOR* 5-9
MSG_HEX: *MONITOR* 5-9
MSG_MED1: *MONITOR* 5-9
MSG_MED2: *MONITOR* 5-9
MSG_MNEM: *MONITOR* 5-9
MSG_OFF: *MONITOR* 5-9
MSG_SHRT: *MONITOR* 5-9
MTEI*: *SIMULATION* 4-21

N

N200: *SIMULATION* 4-9
N200*: *SIMULATION* 4-19
N201: *SIMULATION* 4-9
N201-TX*: *SIMULATION* 4-19
N201*: *SIMULATION* 4-19
N201TX: *SIMULATION* 4-9
N202: *SIMULATION* 4-9
N202*: *SIMULATION* 4-19
NEW_SPEED: *SIMULATION* 4-18
NEW_STATE: *TEST MANAGER* 3-2
NEW_TEI: *SIMULATION* 4-12
NEW_TM: *TEST MANAGER* 2-2
NM20: *SIMULATION* 4-9
NO-ERROR#: *TEST MANAGER* 8-1
NO-IE-ERROR#: *TEST MANAGER* 9-5
NO-MSG#: *TEST MANAGER* 8-1
NORM_MODE: *MONITOR* 6-3
NR: *SIMULATION* 4-7
NS: *SIMULATION* 4-7
NTWK_EMUL: *SIMULATION* 4-14

O

OTHER_EVENT: *TEST MANAGER* 4-9

P

P*SENT: *SIMULATION* 4-19

P0: *SIMULATION* 4-9
P1: *SIMULATION* 4-9
PH_ACTIVATE: *SIMULATION* 4-6
PH_DEACTIVATE: *SIMULATION* 4-6
PIGGY*: *SIMULATION* 4-21
PKT: *SIMULATION* 4-10
PKTS: *SIMULATION* 4-10
PLAY=OFF: *MONITOR* 6-2
PLAY=ON: *MONITOR* 6-2
PLAY_1: *MONITOR* 6-2
PLAY_2: *MONITOR* 6-2
PLAY_3: *MONITOR* 6-2
PLAY_4: *MONITOR* 6-2
PLAY_7: *MONITOR* 6-2
PLAY_8: *MONITOR* 6-2
PLAYBACK: *MONITOR* 6-3
PORT-ID: *SIMULATION* 4-20
POWER_FEED_INVERS: *SIMULATION* 4-17
POWER_FEED_OFF: *SIMULATION* 4-17
POWER_FEED_ON: *SIMULATION* 4-17
PRINT_OFF: *MONITOR* 7-2
PRINT_ON: *MONITOR* 7-2
PT-K: *SIMULATION* 4-8
PT-N201: *SIMULATION* 4-9
PT-N201TX: *SIMULATION* 4-9
PT-T200: *SIMULATION* 4-8
PT-T203: *SIMULATION* 4-8
PTMP: *SIMULATION* 4-14
PTP: *SIMULATION* 4-14

Q

QUIT_COM: *MONITOR* 4-2
QUIT_HELP: *MONITOR* 4-2
QUIT_HEX: *SIMULATION* 3-16
QUIT_MENU: *MONITOR* 4-2
QUIT_TRA: *MONITOR* 7-2

R

R1+ACTIVATED: *MONITOR* 8-1
R1+BPVL: *MONITOR* 8-2
R1+CEPTSPEC: *MONITOR* 8-2
R1+CRCR: *MONITOR* 8-2
R1+DEACTIVATED: *MONITOR* 8-1
R1+FERR: *MONITOR* 8-2
R1+LST_PLOCK: *MONITOR* 8-2
R1+LST_SIGL: *MONITOR* 8-2
R1+LSTFRM: *MONITOR* 8-1
R1+MERR: *MONITOR* 8-2
R1+NORMAL: *MONITOR* 8-2
R1+RECOVERY: *MONITOR* 8-1
R1+RED_ALARM: *MONITOR* 8-2
R1+RI2ERR: *MONITOR* 8-1
R1+UNDEFINED: *MONITOR* 8-1
R1+YEL_ALARM: *MONITOR* 8-2
R1-ACTIVATED: *MONITOR* 8-1
R1-CEPTSPEC: *MONITOR* 8-2
R1-CRCR: *MONITOR* 8-2
R1-DEACTIVATED: *MONITOR* 8-1
R1-FERR: *MONITOR* 8-2
R1-LST_PLOCK: *MONITOR* 8-2
R1-LST_SEGL: *MONITOR* 8-2
R1-LSTFRM: *MONITOR* 8-1
R1-MERR: *MONITOR* 8-2
R1-RECOVERY: *MONITOR* 8-1
R1-RED_ALARM: *MONITOR* 8-2
R1-RI2ERR: *MONITOR* 8-1
R1-UNDEFINED: *MONITOR* 8-1
R1-YEL_ALARM: *MONITOR* 8-2
R1=ALL: *MONITOR* 8-1, 8-2
R1=NONE: *MONITOR* 8-1, 8-2
R1_BPVL: *MONITOR* 8-2
R1_NORMAL: *MONITOR* 8-2
R2+DISC: *MONITOR* 8-3
R2+DM: *MONITOR* 8-3
R2+FRMR: *MONITOR* 8-3
R2+I: *MONITOR* 8-3
R2+INV: *MONITOR* 8-3
R2+REJ: *MONITOR* 8-3
R2+RNR: *MONITOR* 8-3
R2+RR: *MONITOR* 8-3
R2+SABME: *MONITOR* 8-3
R2+UA: *MONITOR* 8-3
R2+UI: *MONITOR* 8-3
R2+XID: *MONITOR* 8-3

R2-DISC: *MONITOR* 8-3
R2-DM: *MONITOR* 8-3
R2-FRMR: *MONITOR* 8-3
R2-I: *MONITOR* 8-3
R2-INV: *MONITOR* 8-3
R2-REJ: *MONITOR* 8-3
R2-RNR: *MONITOR* 8-3
R2-RR: *MONITOR* 8-3
R2-SABME: *MONITOR* 8-3
R2-UA: *MONITOR* 8-3
R2-UI: *MONITOR* 8-3
R2-XID: *MONITOR* 8-3
R2=ALL: *MONITOR* 8-3
R2=NONE: *MONITOR* 8-3
RANGE-ERR#: *TEST MANAGER* 8-1
RC: *SIMULATION* 4-7
RC-ML1*: *SIMULATION* 4-19
RC*: *SIMULATION* 4-19
RC20: *SIMULATION* 4-7
RC20*: *SIMULATION* 4-19
RC201: *SIMULATION* 4-7
RC201*: *SIMULATION* 4-19
RECALL_MESSAGE: *SIMULATION* 6-29
RECORD: *MONITOR* 6-3
RED_BG: *TEST MANAGER* 5-1
REP_OFF: *MONITOR* 5-9
REP_ON: *MONITOR* 5-9
RESER_ALL_IE_MAPS: *SIMULATION* 6-25
RESET_ALL_IE_MAPS: *SIMULATION* 6-11
RESET_IE_MAP: *SIMULATION* 6-11, 6-25
RETURN_STATE: *TEST MANAGER* 3-2
REV_VIDEO: *TEST MANAGER* 5-1
RF_SAPI: *MONITOR* 8-4
RF_SAPI=ALL: *MONITOR* 8-4
RF_TEI: *MONITOR* 8-3
RF_TEI=127: *MONITOR* 8-3
RF_TEI=ALL: *MONITOR* 8-3
RF_TEI127: *MONITOR* 8-3
RTRACE: *TEST MANAGER* 6-1
RUN_SEQ: *TEST MANAGER* 3-2

S

S-DATA-AUTO*: *SIMULATION* 4-13
S_BUS_EXT: *SIMULATION* 4-17
S_BUS_PTP: *SIMULATION* 4-17
S_BUS_SHORT: *SIMULATION* 4-17
S_DISC: *SIMULATION* 4-3
S_DM: *SIMULATION* 4-3
S_FRMR: *SIMULATION* 4-3
S_I: *SIMULATION* 4-3
S_ID_ASSIGNED: *SIMULATION* 4-4
S_ID_CHECK_REQ: *SIMULATION* 4-4
S_ID_CHECK_RESP: *SIMULATION* 4-4
S_ID_DENIED: *SIMULATION* 4-4
S_ID_REMOVE: *SIMULATION* 4-4
S_ID_REQUEST: *SIMULATION* 4-4
S_ID_VERIFY: *SIMULATION* 4-4
S_REJ: *SIMULATION* 4-3
S_REJC: *SIMULATION* 4-3
S_RNR: *SIMULATION* 4-3
S_RNRC: *SIMULATION* 4-3
S_RR: *SIMULATION* 4-3
S_RRC: *SIMULATION* 4-3
S_SABME: *SIMULATION* 4-3
S_UA: *SIMULATION* 4-3
S_UI: *SIMULATION* 4-3
S_XID: *SIMULATION* 4-4
S_XID0: *SIMULATION* 4-4
S_XIDC: *SIMULATION* 4-4
S_XIDC0: *SIMULATION* 4-4
SABME: *SIMULATION* 4-1
SAPI: *SIMULATION* 4-12
SAPI*: *SIMULATION* 4-19
SAVE POOL: *SIMULATION* 3-16
SAVE_MESSAGES: *SIMULATION* 6-29
SAVETS: *TEST MANAGER* 2-2
SCRN_BACK: *MONITOR* 7-2
SCRN_FWD: *MONITOR* 7-2
SECONDES_ELAPSED: *TEST MANAGER* 5-2
SELECT_CR: *SIMULATION* 6-10, 6-24
SEND_FRMR: *SIMULATION* 4-4
SEND_MESSAGE: *SIMULATION* 6-32

SEND_UNIT_MESSAGE: *SIMULATION* 6-32
SENDF: *SIMULATION* 4-3
SENDP: *SIMULATION* 4-1
SEQ{: *TEST MANAGER* 3-2
SET_EVENT_FLAG: *TEST MANAGER* 4-10
SET_INTERFACE: *SIMULATION* 6-12, 6-26
SHIFT-ERR#: *TEST MANAGER* 8-1
SHORT-MSG#: *TEST MANAGER* 8-1
SHUTDOWN_OFF: *MONITOR* 5-10
SHUTDOWN_ON: *MONITOR* 5-10
SI1_BIT=0: *SIMULATION* 4-16
SI1_BIT=1: *SIMULATION* 4-16
SI2_BIT=0: *SIMULATION* 4-16
SI2_BIT=1: *SIMULATION* 4-16
SLAVE_CLK: *SIMULATION* 4-17
SPLIT_OFF: *MONITOR* 5-10
SPLIT_ON: *MONITOR* 5-10
STANDARD_CLK: *SIMULATION* 4-18
START-TIME: *SIMULATION* 4-20
START_LAPSE_TIMER: *TEST MANAGER* 5-2
START_TM_TIMER: *TEST MANAGER* 5-3
STATE{: *TEST MANAGER* 3-1
STATE-L2*: *SIMULATION* 4-19
STATE-TRACE: *TEST MANAGER* 6-1
STATE_INIT{: *TEST MANAGER* 3-2
STOP_ALL_TIMERS: *TEST MANAGER* 5-3
STOP_SCANNING: *SIMULATION* 6-15
STOP_TM_TIMER: *TEST MANAGER* 5-3
STORE_MESSAGE: *SIMULATION* 6-29
STORE_MSG: *SIMULATION* 6-31
STRING->BUFFER: *TEST MANAGER* 5-4
SVC_OFF: *SIMULATION* 6-15, 6-27
SVC_ON: *SIMULATION* 6-15, 6-27

T

T.: *TEST MANAGER* 6-1
T.": *TEST MANAGER* 6-1
T.H: *TEST MANAGER* 6-1
T.TYPE: *TEST MANAGER* 6-1
T+MSG: *MONITOR* 9-4
T-MSG: *MONITOR* 9-4
T200: *SIMULATION* 4-8
T201: *SIMULATION* 4-8
T202: *SIMULATION* 4-8
T203: *SIMULATION* 4-8
TA1: *MONITOR* 9-6
TA2: *MONITOR* 9-6
TA3: *MONITOR* 9-6
TA4: *MONITOR* 9-6
TCLR: *TEST MANAGER* 2-2
TCOLOR: *TEST MANAGER* 6-1
TCR: *TEST MANAGER* 6-1
TEI: *SIMULATION* 4-12
TEMIT: *TEST MANAGER* 6-1
TIME_HOUR: *MONITOR* 5-10
TIME_ON: *MONITOR* 5-10
TIME_SEC: *MONITOR* 5-10
TIME_YEAR: *MONITOR* 5-10
TM.RAM: *TEST MANAGER* 2-3
TM_IPC_SEND: *TEST MANAGER* 5-3
TM_KEYS: *MONITOR* 4-2
TM_RUN: *TEST MANAGER* 2-2
TM_STOP: *TEST MANAGER* 2-2
TM0: *TEST MANAGER* 2-3
TM1: *TEST MANAGER* 2-3
TM2: *TEST MANAGER* 2-3
TM20: *SIMULATION* 4-8
TM3: *TEST MANAGER* 2-3
TM4: *TEST MANAGER* 2-3
TM5: *TEST MANAGER* 2-3
TM6: *TEST MANAGER* 2-3
TM7: *TEST MANAGER* 2-3
TOP: *MONITOR* 7-2
TR1: *MONITOR* 9-2
TR2: *MONITOR* 9-2
TR3: *MONITOR* 9-2
TR4: *MONITOR* 9-2
TRA_ALL: *MONITOR* 7-2
TRA_END: *MONITOR* 7-2
TRA_START: *MONITOR* 7-2
TRANSFER: *MONITOR* 7-2

U

U/N: *SIMULATION* 4-8

UI-TYPE*: *SIMULATION* 4-22
UI_3: *SIMULATION* 4-1
UI_M: *SIMULATION* 4-1
UNEXP-EXTEND#: *TEST MANAGER* 9-5
USER_EMUL: *SIMULATION* 4-14
USR*NET: *SIMULATION* 4-19

V

VA: *SIMULATION* 4-7
VR: *SIMULATION* 4-7
VR*: *SIMULATION* 4-19
VS: *SIMULATION* 4-7
VS*: *SIMULATION* 4-19
VTEI*: *SIMULATION* 4-19

W

W0: *SIMULATION* 4-10
W1: *SIMULATION* 4-10
WAKEUP_OFF: *TEST MANAGER* 2-3
WAKEUP_ON: *TEST MANAGER* 2-3
WD0: *MONITOR* 7-1
WHI_FG: *TEST MANAGER* 5-1
WRITE: *SIMULATION* 3-16

X

X0: *SIMULATION* 4-10
X1: *SIMULATION* 4-10
XID-K*: *SIMULATION* 4-21
XID-RX-N201*: *SIMULATION* 4-21
XID-T200*: *SIMULATION* 4-21
XID-TX-N201*: *SIMULATION* 4-21
XID_OFF: *SIMULATION* 5-3
XID_ON: *SIMULATION* 5-3
XIDC: *SIMULATION* 4-2
XIDV: *SIMULATION* 4-12

Y

Y0: *SIMULATION* 4-10
Y1: *SIMULATION* 4-10
YALM_OFF: *SIMULATION* 4-16
YALM_ON: *SIMULATION* 4-16
YEL_FG: *TEST MANAGER* 5-1

Z

Z0: *SIMULATION* 4-10
Z1: *SIMULATION* 4-10

ISDN LAYER 3 INTERNATIONAL VARIANT (Q931)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version : V2.0
System Version : MF 2.1

Revision Date : January 1990

1 INTRODUCTION 1-1

2 MESSAGE TYPE IDENTIFIER 2-1

3 INFORMATION ELEMENT TYPE IDENTIFIER 3-1

4 OCTET IDENTIFIER 4-1

5 LAYER 3 TIMER 5-1

6 ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1 Shift IE (I#SHIFT) 6-1

6.2 Congestion Level IE (I#CONG_LEVEL) 6-1

6.3 Repeat Indicator IE (I#REP_IND) 6-2

6.4 Segmented Message IE (I#SEG_MSG) 6-2

6.5 Bearer Capability IE (I#BEARER_CAP) 6-3

6.6 Cause IE (I#CAUSE) 6-7

6.7 Call Identity IE (I#CALL_ID) 6-9

6.8 Call State IE (I#CALL_STATE) 6-10

6.9 Channel Identification IE (I#CHANNEL_ID) 6-11

6.10 CCITT-Standardized Facility IE (I#FACIL) 6-12

6.11 Progress Indicator IE (I#PROGREES_ID) 6-15

6.12 Network-Specific Facility IE (I#NETWORK_FACIL) 6-16

6.13 Notification Indicator IE (I#NOTIFIC_IND) 6-16

6.14 Display IE (I#DISPLAY) 6-17

6.15 Date/Time IE (I#DATE) 6-17

6.16 Keypad IE (I#KEYPAD) 6-17

6.17 Signal IE (I#SIGNAL) 6-18

6.18 Switchhook IE (I#SWITCHHOOK) 6-18

6.19 Feature Activation IE (I#FEATURE_ACT) 6-19

6.20 Feature Indication IE (I#FEATURE_IND) 6-19

6.21 Information Rate IE (I#INFO_RATE) 6-19

6.22 End-to-End Transit Delay IE (I#END_DELAY) 6-21

6.23 Transit Delay Selection and Indication IE (I#TRANS_DEL) 6-21

6.24 Packet Layer Binary Parameters IE (I#PACKET_BIN) 6-22
6.25 Packet Layer Window Size IE (I#PACKET_WIN) 6-22
6.26 Packet Size IE (I#PACKET_SIZE) 6-23
6.27 Calling Number IE (I#CALLING_NUM) 6-23
6.28 Calling Subaddress IE (I#CALLING_SAD) 6-24
6.29 Called Number IE (I#CALLED_NUM) 6-24
6.30 Called Subaddress IE (I#CALLED_SAD) 6-25
6.31 Redirecting Number IE (I#REDIRECTING_NUM) 6-25
6.32 Transit Network Selection IE (I#TRANS_NW_SEL) 6-26
6.33 Restart Indicator IE (I#RESTART_IND) 6-27
6.34 Low Layer Compatability IE (I#LOW_LAY_COMP) 6-28
6.35 High Layer Compatability IE (I#HI_LAY_COMP) 6-33
6.36 User User Information IE (I#UU_INFO) 6-34
6.37 Escape for Extension IE (I#ESCAPE) 6-34

1 INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2 MESSAGE TYPE IDENTIFIER

Message Identifier	CCITT Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#CON_CON	Congestion Control
M#FAC	Facility
M#INFO	Info
M#NOTIFY	Notify
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry
M#SEGMENT	Segmented Message

3 INFORMATION ELEMENT TYPE IDENTIFIER

IE Identifier	CCITT Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#REP_IND	Repeat Indicator
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#SWITCHHOOK	Switchhook
I#FEATURE_ACT	Feature Activation
I#FEATURE_IND	Feature Indication
I#INFO_RATE	Information Rate
I#END_DELAY	End-to-End Transit Delay
I#TRANS_DEL	Transit Delay Selection and Indication
I#PACKET_BIN	Packet Binary Parameters
I#PACKET_WIN	Packet Binary Window Size
I#PACKET_SIZE	Packet Size
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#TRANS_NW_SEL	Transit Network Selection
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information
I#ESCAPE	Escape for Extension

4 OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,	OCTET_3.1,	OCTET_3.2,
OCTET_3.3,				
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,	OCTET_6.1,	OCTET_6.2,	OCTET_6.3,
OCTET_6.4,	OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5 LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T301	#T301	180	180
Timer T302	#T302	15	15
Timer T303	#T303	4	4
Timer T304	#T304	15	20
Timer T305	#T305	30	30
Timer T306	#T306	30	30
Timer T307	#T307	180	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	10	10
Timer T312	#T312	6	6
Timer T313	#T313	4	4
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	60	60
Timer T318	#T318	4	4
Timer T319	#T319	4	4
Timer T320	#T320	30	30
Timer T321	#T321	30	30
Timer T322	#T322	4	4
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6 ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1 Shift IE (I#SHIFT)

Selector Name and related Constants	CCITT Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	Codeset Specified in Shift IE
#CODESET0	*I.451 (Q.931) IE*
#CODESET5	*National Use IE*
#CODESET6	*Local Network Specific IE*
#CODESET7	*User Specific IE*

6.2 Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	CCITT Reference
->CL_CONGESTION_LEVEL	Congestion Level
#RECEIVER_READY	*Receiver Ready*
#RECEIVER_NOT_READY	*Receiver not Ready*

6.3 Repeat Indicator IE (I#REP_IND)

Selector Name and related Constants	CCITT Reference
->RP_INDICATION #PRIORITIZED	Repeat Indication *Prioritized List*

6.4 Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.5 Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_7

Selector Name and related Constants	CCITT Reference
->BC_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for present network*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S #2x64KBIT/S #384KBIT/S #1536KBIT/S #1920KBIT/S	Information Transfer Rate, Octet 4 *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
->BC_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNCONSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->BC_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->BC_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->BC_SYMMETRY #BIDIRECT_SYMMENTRIC	Symmetry, Octet 4a *Bidirectional Symmetric*

Selector Name and related Constants	CCITT Reference
->BC_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->BC_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.7XX 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->BC_SYNC/ASYNC	Synchronous or Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->BC_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->BC_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6 and X.1*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*

Selector Name and related Constants	CCITT Reference
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation V.6 and X.1*
#1.2/0.075KBIT/S	*1.2/0.075 kbit/s Recommendation V.6 and X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.75KBIT/S	*0.75 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->BC_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*not used*
#INT_8KBIT/S	*8 kbit/s*
#INT_16KBIT/S	*16 kbit/s*
#INT_32KBIT/S	*32 kbit/s*
->BC_NIC_ON_TX	Network internal clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->BC_NIC_ON_RX	Network internal clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->BC_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*No Flow Control required*
#REQUIRED	*Flow Control required*
->BC_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept Flow Control*
#ACCEPT	*Can accept Flow Control*
->BC_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->BC_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->BC_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*
->BC_LLI_NEG	Logical Link negotiation, Octet 5b.2
#DEFAULT	*Default, LLI=256 only*
#FULL	*Full protocol negotiation*

Selector Name and related Constants	CCITT Reference
->BC_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assigner'* *Originator is 'Assigner only'*
->BC_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out-band negotiation, Octet 5b.2 *Negotiation is done 'Out-band'* *Negotiation is done 'In-band'*
->BC_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of stop bits, Octet 5c *Not used* *1 bit* *1.5 bits* *2 bits*
->BC_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of data bits, Octet 5c *Not used* *5 bits* *7 bits* *8 bits*
->BC_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity information, Octet 5c *odd* *even* *none* *forced to 0* *forced to 1*
->BC_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
->BC_MODEM_TYPE	Modem type, Octet 5d
->BC_LAYER2_ID #LAYER2	Layer Identification, Octet 6 *User Info Layer 2 Protocol*
->BC_L2_PROTOCOL #Q.921 #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation Q.921/I.441* *Recommendation X.25, Link Layer*
->BC_LAYER3_ID #LAYER3	Layer 3 Identification, Octet 7 *User Info Layer 3 Protocol*
->BC_L3_PROTOCOL #Q.931 #X.25_PACKET	Layer 3 Protocol Identifier, Octet 7 *Recommendation Q.921/I.441* *Recommendation X.25, Packet Layer*

Note 1 : Octet 5b.1 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V120

6.6 Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_5

Selector Name and related Constants	CCITT Reference
->C_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->C_LOCATION #USER #LOACAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Transit Network* *Remote Public Network* *Remote Private Network* *International Network* *Network Beyond Interworking Point*
->C_RECOMMENDATION #REC_Q.931 #REC_X.21 #REC_X.25	Recommendation, Octet 3a *Recommendation Q.931* *Recommendation X.21* *Recommendation X.25*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER #NO_ROUTE_TO_TRANSIT #NO_ROUTE_TO_DEST #CHANNEL_UNACCEPTABLE #CALL_AWARDED	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned Number* *No Route to Transit Network* *No Route to Destination* *Channel Unacceptable* *Call awarded*
->C_CAUSE_CLASS #NORMAL1 ->C_CAUSE_VALUE #NORMAL_CLEARING #USER_BUSY #NO_USER_RESPOND #NO_ANSWER #CALL_REJECTED #NUMBER_CHANGED #NON_SELECTED_CLEAR	Cause Class, Octet 4 *Normal Event Class 1* Cause Value, Octet 4 *Normal call clearing* *User Busy* *No User Responding* *User alerting, no answer* *Call Rejected* *Number Changed* *Non selected user clearing*

Selector Name and related Constants	CCITT Reference
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility Rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
->C_CAUSE_CLASS	Cause Class, Octet 4
#RESOURCE_UNAVAIL	*Resource not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NO_CHANNEL_AVAIL	*No Circuit/Channel Available*
#OUT_OF_ORDER	*Network Out of Order*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access Information Discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#QUALITY_UNAVAIL	*Quality of service unavailable*
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer Capability not Authorized*
#BEARER_UNAVAIL	*Bearer Capability not Available*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#CHANNEL_TYPE_UNIMPL	*Channel Type not Implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#ONLY_RESTRICTED	*Restricted only*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INVALID_MESSAGE	*Invalid Message*
->C_CAUSE_VALUE	Cause Value, Octet 4
#INVALID_CALL_REF	*Invalid Call Reference Value*
#CHANNEL_NONEXISTANT	*Channel not Existant*
#CALL_ID_NONEXISTANT	*Call Identity does not Exist*
#CALL_ID_IN_USE	*Call Identity in Use*
#NO_CALL_SUSPENDED	*No call suspended*
#CALL_CLEARED	*Call has been cleared*
#INCOMPATIBLE_DEST	*Incompatible Destination*
#INV_TRANS_NET	*Invalid transit network selection*
#INVALID_MESSAGE_UNSPEC	*Invalid Message*

Selector Name and related Constants	CCITT Reference
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on Timer Expiry* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.7 Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.8 Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->CS_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*
#CALL_ABORT	*Call Abort*
#OVERLAP_RECEIVING	*Overlap Receiving*
#RESTART_REQUEST	*Restart request*
#RESTART	*Restart*

6.9 Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_3.3

Selector Name and related Constants	CCITT Reference
->CID_INT_PRESENT #EXPLICIT #IMPLICIT	Interface Identifier Present, Octet 3 *Explicitly Identified* *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Other Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #D_CHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *D-Channel Indentified* *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_INT_ID	Interface identifier, Octet 3.1 *
->CID_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3.2 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 3.2 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS #H0_CHANNEL_UNITS #H11_CHANNEL_UNITS #H12_CHANNEL_UNITS	Channel or Map Type, Octet 3.2 *B-Channel Units* *H0-Channel Units* *H11-Channel Units* *H12-Channel Units*
->CID_MAP	Slot Map, Octet 3.3
->CID_NUMBER	Channel number, Octet 3.3

6.10 CCITT-Standardized Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	CCITT Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	CCITT Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #NO_SUPPORT	Error Value, Octet 6.5 (Note 3) *Not supported*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	CCITT Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.11 Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PI_CODING_STANDARD	Coding Standard, Octet 3.2
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->PI_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVAT	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->PI_DESCRIPTION	Progress Description, Octet 4
#NOT_END_TO_END	*Call is not end-to-end ISDN*
#DEST_NON_ISDN	*Destination Address is non-ISDN*
#ORIG_NON_ISDN	*Origination Address is non-ISDN*
#RETURNED_TO_ISDN	*Call has Returned to the ISDN*
#INBAND_INFO_AVAIL	*In-band Info now Available*

6.12 Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_4

Selector Name and related Constants	CCITT Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3.1 *User specified* *National network id.* *International network id.*
->NF_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3.1 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->NF_NET_ID	Network Identification, Octet 3.2 *
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.13 Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change*

6.14 Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->D_DISPLAY	Display, Octet 3 *

6.15 Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8

Selector Name and related Constants	CCITT Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7
->DT_SECOND	Second, Octet 8

6.16 Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->K_KEYPAD	Keypad, Octet 3 *

6.17 Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*

6.18 Switchhook IE (I#SWITCHHOOK)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->SW_VALUE	Switchhook Value, Octet 3
#ON_HOOK	*On-hook*
#OFF_HOOK	*Off-hook*

6.19 Feature Activation IE (I#FEATURE_ACT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A

Selector Name and related Constants	CCITT Reference
->FA_ID_NUMBER	Feature identifier number, Octet 3
->FA_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a

6.20 Feature Indication IE (I#FEATURE_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	CCITT Reference
->FI_ID_NUMBER	Feature identifier number, Octet 3
->FI_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a
->FI_STATUS_IND #DEACTIVATED #ACTIVATED #PROMPT #PENDING	Status indicator, Octet 4 *Feature is inactive* *Feature is active* *Feature prompt* *Feature is pending*

6.21 Information Rate IE (I#INFO_RATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6

Selector Name and related Constants	CCITT Reference
->IR_INCOMING #75_B/S #150_B/S #300_B/S #600_B/S #1200_B/S #2400_B/S #4800_B/S #9600_B/S #19200_B/S #48000_B/S	Incoming information rate, Octet 3 *75 bit/s* *150 bit/s* *300 bit/s* *600 bit/s* *1200 bit/s* *2400 bit/s* *4800 bit/s* *9600 bit/s* *19200 bit/s* *48000 bit/s*

Selector Name and related Constants	CCITT Reference
->IR_INCOMING	Incoming information rate, Octet 3
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_OUTGOING	Outgoing information rate, Octet 4
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_MIN_INCOMING	Minimum incoming info rate, Octet 5
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_MIN_INCOMING	Minimum incoming info rate, Octet 5
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*

6.22 End-to-End Transit Delay IE (I#END_DELAY)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B

Selector Name and related Constants	CCITT Reference
->ED_CUM_DELAY	Cumultative transit delay value, Octet 3, 3a, 3b
->ED_CUM_DELAY_3	Cumultative trans. del. val., bit 16-15 (Octet 3)
->ED_CUM_DELAY_3A	Cumultative trans. del. val., bit 14-8 (Octet 3a)
->ED_CUM_DELAY_3B	Cumultative trans. del. val., bit 7-1 (Octet 3b)
->ED_REQ_DELAY	Requested transit delay value, Octet 4, 4a, 4b
->ED_REQ_DELAY_4	Requested trans. del. val., bit 16-15 (Octet 4)
->ED_REQ_DELAY_4A	Requested trans. del. val., bit 14-8 (Octet 4a)
->ED_REQ_DELAY_4B	Requested trans. del. val., bit 7-1 (Octet 4b)
->ED_MAX_DELAY	Maximum transit delay value, Octet 5, 5a, 5b
->ED_MAX_DELAY_5	Maximum trans. del. val., bit 16-15 (Octet 5)
->ED_MAX_DELAY_5A	Maximum trans. del. val., bit 14-8 (Octet 5a)
->ED_MAX_DELAY_5B	Maximum trans. del. val., bit 7-1 (Octet 5b)

6.23 Transit Delay Selection and Indication IE (I#TRANS_DEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B

Selector Name and related Constants	CCITT Reference
->TD_SEL/IND	Transit delay sel/ind value, Octet 3, 3a, 3b
->TD_SEL/IND_3	Transit delay sel/ind val., bit 16-15 (Octet 3)
->TD_SEL/IND_3A	Transit delay sel/ind val., bit 14-8 (Octet 3a)
->TD_SEL/IND_3B	Transit delay sel/ind val., bit 7-1 (Octet 3b)

6.24 Packet Layer Binary Parameters IE (I#PACKET_BIN)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->PB_FAST_SELECT #NOT_REQUESTED_0 #NOT_REQUESTED_1 #REQUESTED_NOT_REST #REQUESTED_REST	Fast select, Octet 3 *Not requested* *Not requested* *Requested, no restriction* *Requested, restriction*
->PB_EXP_DATA #NO_REQUEST #REQUEST_IND	Expedited data, Octet 3 *No request / request denied* *Request indicated / accepted*
->PB_DELIVERY_CONF #LINK_BY_LINK #END_TO_END	Delivery confirmation, Octet 3 *Link-by-link confirmation* *End-to-end confirmation*
->PB_MODULUS #MODULUS_8 #MODULUS_128	Modulus, Octet 3 *Modulus 8 sequencing* *Modulus 128 sequencing*

6.25 Packet Layer Window Size IE (I#PACKET_WIN)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PWS_FORWARD	Forward value, Octet 3
->PWS_BACKWARD	Backward value, Octet 4

6.26 Packet Size IE (I#PACKET_SIZE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PS_FORWARD	Forward value, Octet 3
->PS_BACKWARD	Backward value, Octet 4

6.27 Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber Number* *Abbreviated Number*
->CLGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*
->CLGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED #NUMBER_UNAVAIL	Presentation Indicator, Octet 3 *Presentation Allowed* *Presentation Restricted* *Number Not Available Due to Interworking*
->CLGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.28 Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.29 Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLDN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber Number* *Abbreviated Number*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.30 Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLDS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.31 Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	CCITT Reference
->RDGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber number* *Abbreviated Number*
->RDGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*

Selector Name and related Constants	CCITT Reference
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->RDGN_REASON #CALL_FORW_BUSY #CALL_FORW_NO_REPLY #CALLED_OUT_OF_ORDER #FORW_BY_CALLED #UNCOND_OR_SYS	Reason for redirection, Octet 3b *Call forwarding or called DTE busy* *Call forwarding no reply* *Called DTE out of order* *Forwarding by called DTE* *Unconditional or systematic red.*
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.32 Transit Network Selection IE (I#TRANS_NW_SEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->TR_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3 *User specified* *National network id.* *International network id.*
->TR_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->TR_NET_ID	Network Identification, Octet 4 *

6.33 Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->RI_CLASS #INDICATED_CHANNEL #SINGLE_INTERFACE #ALL_INTERFACES	Class of Channel or Interface, Octet 3 *Indicated Channel* *Single Interface* *All Interface*

6.34 Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	CCITT Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 2.1 or 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation indicator, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S #2x64KBIT/S #384KBIT/S #1536KBIT/S #1920KBIT/S	Information Transfer Rate, Octet 4 *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*

Selector Name and related Constants	CCITT Reference
->LL_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->LL_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->LL_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#NO_ADD_L1_PROTOCOL	*No Additional Layer 1 Protocol*
#RATE_ADAPTION	*Rate Adaption*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.711 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*

Selector Name and related Constants	CCITT Reference
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.75KBIT/S	*0.75 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#INT_NOT_USED	*Not Used*
#INT_8KBIT/S	*8kbit/s*
#INT_16KBIT/S	*16kbit/s*
#INT_32KBIT/S	*32kbit/s*
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*Flow Control not required*
#REQUIRED	*Flow Control Required*
->LL_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot Accept Flow Control*
#ACCEPT	*Can Accept Flow Control*
->LL_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->LL_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->LL_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*

Selector Name and related Constants	CCITT Reference
->LL_LLI_NEG #DEFAULT #FULL	Logical Link negotiation, Octet 5b.2 *Default, LLI=256 only* *Full protocol negotiation*
->LL_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assigner'* *Originator is 'Assigner only'*
->LL_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out-band negotiation, Octet 5b.2 *Negotiation is done 'Out-band'* *Negotiation is done 'In-band'*
->LL_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of Stop Bits, Octet 5c *Not used* *1 Bit* *1.5 Bits* *2 Bits*
->LL_DATA_BITS #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of Data Bits, Octet 5c *5 Bits* *7 Bits* *8 Bits*
->LL_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity Information, Octet 5c *Odd* *Even* *None* *Forced to 0* *Forced to 1*
->LL_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
->LL_MODEM_TYPE	Modem type, Octet 5d
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #ISO_1745 #Q.921 #X.25_LINK #X.25_MULTILINK #EXTENDED_LAPB #HDLC_ARM #HDLC_NRM #HDLC_ABM #LAN_LLC #X.75	Layer 2 Protocol Identifier, Octet 6 *Basic mode ISO 1745* *Recommendation Q.921 (i.441)* *Recommendation X.25, Link level* *Recommendation X.25 Multilink* *Extended LAPB; half duplex (T.71)* *HDLC ARM (ISO 4335)* *HDLC NRM (ISO 4335)* *HDLC ABM (ISO 4335)* *LAN Logical link control (ISO 8802/2)* *Rec. X.75, single link procedure*

Selector Name and related Constants	CCITT Reference
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET #ISO_8208 #ISO_8348 #ISO_8473 #T.70	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer* *ISO 8208* *ISO 8348* *ISO 8473* *Rec. T.70, minimum network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.35 High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A

Selector Name and related Constants	CCITT Reference
->HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->HL_INTERPRETATION	Interpretation, Octet 3
#FIRST_CHARACTER	*First HL Characteristics to be Used*
->HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
->HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#MAINTENANCE	*Reserved for Maintenance (Note 1)*
#MANAGEMENT	*Reserved for Management (Note 1)*
->HL_CHARACTER_EXT	Extended High Layer Characteristics, Octet 4a
#TELEPHONY	*Telephony*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application*
#NOT_AVAIL1	*Not available for assignment*
#NOT_AVAIL2	*Not available for assignment*

Note 1 : Only in this case, Octet 4a may follow

6.36 User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->UU_PROTOCOL_DISCR	Protocol discriminator, Octet 3
#USER_SPECIFIC_PD	*User specific protocol*
#OSI_HIGHER_LAYER	*OSI Higher Layer Protocols*
#X.244	*X.244*
#SYS_MNG_CONVERG	*System mng. convergence function*
#IA5_CHAR	*IA5 characters*
#V.120	*reserved for V.120 rate adaption*
#Q.931_CALL_CONTROL	*Q931 Call Control Messages*
->UU_USER_INFO	User-User Information, Octet 4 *

6.37 Escape for Extension IE (I#ESCAPE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->EX_IE_ID	Information element identifier, Octet 3
->EX_ESCAPE	Contents of information element, Octet 4 *

ISDN LAYER 3 NATIONAL VARIANT (1TR6)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.1 Rev.0
System Version	:	MF 2.2 Rev. 1
Revision Date	:	January 1993
Engineering	:	AUT E33 D / SteHu
Quality Control	:	MWB KQP 2 HBR

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

2.1. MESSAGES WITH PD N1 2-1

2.2. MESSAGES WITH PD N0 2-2

3. W-ELEMENT IDENTIFIER 3-1

3.1. CODESET O 3-1

3.2. CODESET 6 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING W-ELEMENT FIELD VALUES 6-1

6.1. NATIONAL CS O IES (W-ELEMENTS) 6-1

6.1.1. SHIFT (I#SHIFT) 6-1

6.1.2. CONGESTION LEVEL (I#CONG_LEVEL) 6-1

6.1.3. CAUSE (I#CAU) 6-2

6.1.4. CONNECTED ADDRESS (I#CAD) 6-3

6.1.5. CALL IDENTITY (I#CAI) 6-3

6.1.6. CHANNEL ID (I#CHI) 6-4

6.1.7. NETWORK SPECIFIC FACILITIES (I#NSF) 6-5

6.1.7.1. PARAMETER FIELD FOR FACILITY CODE #SPERRE 6-8

6.1.7.2. PARAMETER FIELD FOR FAC. CODE #UMLEITUNG_STAENDIG/#UMLEITUNG.FALLWEISE 6-8

6.1.7.3. PARAMETER FIELD FOR FAC. CODE #B_KANAL_UEBERNEHMEN 6-9

6.1.7.4. PARAMETER FIELD FOR FAC. CODE #DIENSTWECHSEL_1/#DIENSTWECHSEL_2 6-9

6.1.7.5. PARAMETER FIELD FOR FAC. CODE #GBG 6-9

6.1.8. DISPLAY (I#DSP) 6-10

6.1.9. KEYPAD (I#KEY) 6-10

6.1.10. ORGINATION ADDRESS (I#OAD) 6-10

6.1.11. DESTINATION ADDRESS (I#DAD) 6-11

6.1.12. USER-USER INFORMATION (I#UUI) 6-11

6.2. NATIONAL CS 6 IES (W-ELEMENTS) 6-12

6.2.1. SERVICE INDICATOR (I#SIN) 6-12

6.2.2. CHARGING INFO (I#CIF) 6-14

6.2.3. DATE (I#DTE) 6-14

6.2.4. FACILITY SELECT (I#FSE) 6-15

6.2.5. STATUS OF FACILITIES (I#SOF) 6-15

6.2.6. STATUS DES GERUFENEN TLN (I#SDT) 6-16

6.2.7. ADDITIONAL TRANSMISSION ATTRIBUTES (I#ATA) 6-16

Program:	1TR6 (National variant of the ISDN layer 3 protocol)
Software Version:	V3.1
Previous Version:	V3.0
System Version:	MF 2.2.1
Revision Date:	January 1993

☞ *Please note that it is important to read this paper before reading the manual, because some*
☞ *changed functions are described there.*

CHANGES MADE:

- Updated up to 'Berichtigung Nr. 7' of the 1TR6 protocol.

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

2.1. MESSAGES WITH PD N1

Message Identifier	1TR6 Reference
M#ALERT	Alerting
M#CALL_SENT	Call Sent
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#SETUP	Setup
M#SETUP_ACK	Setup Acknowledge
M#RES	Resume
M#RES_ACK	Resume Acknowledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknowledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DET	Detach
M#DISC	Disconnected
M#REL	Release
M#REL_ACK	Release Acknowledge
M#CANC_ACK	Cancel Acknowledge
M#CANC_REJ	Cancel Reject
M#CON_CON	Congestion Control
M#FAC	Facility
M#FAC_ACK	Facility Acknowledge
M#FAC_CANC	Facility Cancel
M#FAC_REG	Facility Register
M#FAC_REJ	Facility Reject
M#INFO	Information
M#REG_ACK	Register Acknowledge
M#REG_REJ	Register Reject
M#STATUS	Status

2.2. MESSAGES WITH PD N0

Message Identifier	1 TR6 Reference
M#REG_IND	Register Indication
M#CANC_IND	Cancel Indication
M#FAC_STA	Facility Status
M#STA_ACK	Status Acknowledge
M#STA_REJ	Status Reject
M#FAC_INF	Facility Information
M#INF_ACK	Information Acknowledge
M#INF_REJ	Information Reject
M#CLOSE	Close
M#CLO_ACK	Close Acknowledge

3. W-ELEMENT IDENTIFIER

3.1. CODESET 0

W-Element	1TR6 Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#CONG_LEVEL	Congestion Level
I#CAU	Cause
I#CAD	Connected Address
I#CAI	Call Identity
I#CHI	Channel Identification
I#NSF	Network Specific Facilities
I#DSP	Display
I#KEY	Keypad
I#OAD	Orgination Address
I#DAD	Destination Address
I#UUI	User-User Information

3.2. CODESET 6

W-Element	1TR6 Reference
I#SIN	Service Indicator
I#CIF	Charging Information
I#DTE	Date
I#FSE	Facility Select
I#SOF	Status of Facility
I#SDT	Status of the called address
I#ATA	Additional transmission attributes

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements ("W-Elements") :

OCTET_3,	**OCTET_4,**	**OCTET_4.1**	**OCTE_4.2**	**OCTET_5,**
OCTET_6,	**OCTET_7,**	**OCTET_8,**	**OCTET_9,**	**OCTET_10**
OCTET_12,	**OCTET_13,**	**OCTET_14.**		

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds) network side	user side P-MP	user side P-P
Timer T302	#T302	12	-	12
Timer T303	#T303	1,5	4	4
Timer T304	#T304	-	20	60
Timer T305	#T305	120	4	30
Timer T307	#T307	120	-	-
Timer T308	#T308	4	4	4
Timer T310	#T310	-	120	70
Timer T313	#T313	-	4	4
Timer T318	#T318	-	4	-
Timer T319	#T319	-	4	-
Timer T3AA	#T3AA	120	-	120
Timer T3AB	#T3AB	15	-	-
Timer T3AC	#T3AC	20	-	-
Timer T3AD	#T3AD	30	-	-
Timer T3AE	#T3AE	180	-	-
Timer T3AF	#T3AF	15 min	-	-
Timer T3AG	#T3AG	5	-	-
Timer T3D1	#T3D1	-	10	10
Timer T3D2	#T3D2	-	4	4

6. ACCESSING W-ELEMENT FIELD VALUES

The following sections contain all information for access of the Information Element parameter list including relevant octet identifier for all Information Elements,selector names associated with each Information Element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. NATIONAL CS O IES (W-ELEMENTS)

6.1.1. SHIFT (I#SHIFT)

Selector Name and related Constants	1 TR6 Reference
->SH_TYPE #NON_LOCKING #LOCKING	Locking/Non-locking Shift Type *Non Locking* *Locking*
->SH_CODESET #CODESET0 #CODESET6 #CODESET7	New codeset identification *Codesatz 0* *Codesatz 6* *Codesatz 7*

6.1.2. CONGESTION LEVEL (I#CONG_LEVEL)

Selector Name and related Constants	1 TR6 Reference
->CL_CONGESTION_LEVEL #RECEIVER_READY #RECEIVER_NOT_READY	Congestion Level *Receiver Ready* *Receiver not Ready*

6.1.3. CAUSE (I#CAU)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 REFERENCE
->CAU_VALUE	Cause value, Octet 3
1TR6#INVALID_CR	*Invalid call reference value*
1TR6#BEARER_SERVICE_UNIMPL	*Bearer service not implemented*
1TR6#CALL_ID_NONEXISTENT	*Call identity does not exist*
1TR6#CALL_ID_IN_USE	*Call identity in use*
1TR6#NO_CHANNEL_AVAIL	*No Channel available*
#FACILITY_UNIMPL	*Requested facility not implemented*
#FACILITY_NOT_SUBSCRIBED	*Requested facility not subscribed*
#OUTGOING_CALL_BARRED	*Outgoing calls barred*
#USER_ACCESS_BUSY	*User access busy*
#NEG_GBG_VGL	*Connection not possible (neg. CUG comp.)*
#KOM_SPV_NOTALLOWED	*Connection not possible (neg. call number check)*
#NOT_OBTAINABLE	*Destination not obtainable*
1TR6#NUMBER_CHANGED	*Number changed*
1TR6#OUT_OF_ORDER	*Out of order*
#NO_RESPONSE	*No user responding*
1TR6#USER_BUSY	*User busy*
#INCOMING_BARRED	*Incoming calls barred*
1TR6#CALL_REJECTED	*Call rejected*
#NTWK_CONGESTION	*Network congestion*
#REMOTE_INITIATED	*Remote user initiated*
#LOC_PROC_ERR	*Local procedure error*
#REM_PROC_ERR	*Remote procedure error*
#REM_SUSPENDED	*Remote user suspended*
#REM_RESUMED	*Remote user resumed*
#USER_INFO_DISCARDED	*User info discarded locally*
#NONEXISTENT_CUG	*Non existent CUG*
->CAU_CODING_STANDARD	Coding standard, Octet 4
#NATIONAL	*National standard*
->CAU_LOCATION	Location, Octet 4
#PRIVATE	*Private network*
#PUBLIC	*Public network*
#NO_INFO	*No information*

6.1.4. CONNECTED ADDRESS (I#CAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->CAD_ADDR_TYPE **#UNKNOWN** **#INTERNATIONAL** **#NATIONAL**	Type of address, Octet 3 *Unknown* *International Number* *National Number*
->CAD_ADDR_PLAN **#UNKNOWN_PLAN** **#ISDN_PLAN**	Numbering plan ID, Octet 3 *Unknown* *ISDN Numbering Plan E. 164*
->CAD_ADDRESS	Address (max. length 16 Bytes) Octet 4 *

6.1.5. CALL IDENTITY (I#CAI)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->CAI_ID (any pattern allowed)	Call identity, Octet 3

6.1.6. CHANNEL ID (I#CHI)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	1 TR6 Reference
->CHI_PAT1 **#CHI_BA** **#CHI_PRI**	Bits 5-8 in Octet 3 : *Valid pattern for Basic Access* *Valid pattern for Primary Rate*
->CHI_P/E **#PREFERRED** **#EXCLUSIVE**	Preferred / exclusive bit, Octet 3 *Preferred* *Exclusive*
->CHI_PAT2 **#CHI_PAT2**	Bit 3 in Octet 3 *Valid pattern bit 3 in octet 3*
->CHI_ICS BASIC ACCESS: **#NO_CHANNEL** **#B1_CHANNEL** **#B2_CHANNEL** **#ANY_CHANNEL** PRIMARY RATE : **#NO_CHANNEL** **#SEE_CHAN_NO** **#ANY_CHANNEL**	Info Channel Selection, Octet 3 *Kein Kanal* *B1 Kanal* *B2 Kanal* *Beliebiger Kanal* *Kein Kanal* *Siehe "Channel Number"* *Beliebiger Kanal*
->CHI_PAT3 **#CHI_PAT3**	Bits 1 - 8 in Octet 4 (PRI only) *Valid pattern*
->CHI_CHANNEL	Channel number, Octet 5 (PRI only)

6.1.7. NETWORK SPECIFIC FACILITIES (I#NSF)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8, OCTET_9, OCTET_10, OCTET_12, OCTET_13, OCTET_14

Selector Name and related Constants	1 TR6 Reference
->NSF_PAT	Bits 1 - 8 in Octet 3
#NSF_PAT	*Valid pattern*
->NSF_CODE	facility code, Octet 4
#SPERRE	*Sperre*
#UMLEITUNG_STAENDIG	*Anrufumleitung (ständig)*
#UMLEITUNG_FALLWEISE	*Anrufumleitung (fallweise)*
#KONFERENZ	*Konferenz*
#B_KANAL_UEBERNEHMEN	*B-Kanal übernehmen*
#HALT_AKTIVIEREN	*Aktivieren einer gehaltenen Verbindung*
#DREIERKONFERENZ	*Dreierkonferenz*
#DIENSTWECHSEL_1	*einseitiger Dienstwechsel*
#DIENSTWECHSEL_2	*zweiseitiger Dienstwechsel*
#NR_IDENTIFIZIERUNG	*Rufnummer-Identifizierung*
#GBG	*Geschlossene Benutzergruppe*
#ANZ_UEBERGEBENER_RUF	*Anzeige: übergebener Ruf*
#ANZ_UMLEITUNG	*Anzeige: umgel./weitergesch. Ruf*
#ANZ_UNTERDRUECKEN	*Unterdrücken A-Rufnummer*
#VERB_DEAKTIVIEREN	*Nutzung d. Verbindung deaktivieren*
#VERB_AKTIVIEREN	*Nutzung d. Verbindung aktivieren*
#SEMIPERMANENT	*Semipermanente Verbindung (SPV)*
#RUECKWECHSEL	*Rückwechsel bei zweiseit. D-Wechsel*
#UMLEITUNG_PRIVAT	*Anrufumleitung im privaten Netz*

->NSF_SERVICE	Service, Octet 5
#UNKNOWN	*Gilt für alle Dienste / irrelevant*
#FERNSPRECHEN	*Fernsprechen*
->NSF_ADD_INFO	Additional Information, Octet 6
#ISDN_3.1KHZ	*ISDN-Fernsprechen 3,1 kHz*
#ANALOG	*Fernsprechen analog*
#ISDN_7KHZ	*ISDN-Fernsprechen 7 kHz*
#A/B-DIENSTE	*a/b-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#FAX_GRUPPE_2	*Fax. Gruppe 2*
#FAX_GRUPPE_3	*Fax. Gruppe 3*
#MODEM_DATEN	*Daten über Modem*
#BTX_MODEM	*Btx über Btx-Modem*
#X.21-DIENSTE	*X.21-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#UC_4	*User Class 4*
#UC_5	*User Class 5*
#UC_6	*User Class 6*
#UC_19	*User Class 19*
#FAX_GRUPPE_4	*Telefax Gruppe 4*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BTX_64KBIT/S	*Btx 64 kBit/s*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#DUE_64KBIT/S	*Datenübertragung 64 kBit/s*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
	Nur relevant bei Facility-Code Zweiseitiger/Einseitiger Dienstwechsel und wird dann codiert wie "Additional Information" in "Service Indicator"
#X.25-DIENSTE	*X.25-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#UC_8	*User Class 8*
#UC_9	*User Class 9*
#UC_10	*User Class 10*
#UC_11	*User Class 11*
#UC_13	*User Class 13*
#19.2KBIT/S	*19,2 kbit/s*

#TELETEX_64KBIT/S	*Teletex 64*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#MIXED_MODE_1TR6	*Mixed Mode*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#FERNWIRKEN	*Fernwirken*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#GRAFIKTELEFON	*Grafiktelefondienste*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BILDSCHIRMTEXT	*Bildschirmtext (neuer Standard)*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BILDTELEFON	*Bildtelefon*
->NSF_ADD_INFO	Additional Information, Octet 6
#TON_3.1KHZ	*Ton 3,1 kHz*
#TON_7KHZ	*Ton 7 kHz*
#BILD	*Bild*

6.1.7.1. PARAMETER FIELD FOR FACILITY CODE #SPERRE

Selector Name and related Constants	1 TR6 Reference
->NSF_AS #SPERR_ALLE #SPERR_FERN #SPERR_AUSL #SPERR_INTERKONT	Art der Sperre, Octet 7 *Sperre aller abgehender Verbindungen* *Sperre abghd. Fernverbindungen* *Sperre abghd. Auslandsverbindungen* *Sperre abghd. Interkontinental-verbindungen*
->NSF_AS2 **Parameter wie bei "->NSF_AS"**	Art der Sperre kann mehrfach auftreten, Octet 8
->NSF_AS3 **Parameter wie bei "->NSF_AS"**	Art der Sperre, Octet 9
->NSF_AS4 **Parameter wie bei "->NSF_AS"**	Art der Sperre, Octet 10

6.1.7.2. PARAMETER FIELD FOR FAC. CODE #UMLEITUNG_STAENDIG / #UMLEITUNG.FALLWEISE

Selector Name and related Constants	1 TR6 Reference
->NSF_ADDR_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL	Address number type, Octet 7 *Unknown* *International Number* *National Number*
->NSF_ADDR_PLAN #UNKNOWN_PLAN #ISDN_PLAN	Address numbering plan, Octet 7 *Unknown* *ISDN Numbering Plan E.164*
->NSF_ADDRESS	Address, Octet 8 * (max. length 16 Bytes)

6.1.7.3. PARAMETER FIELD FOR FAC. CODE #B_KANAL_UEBERNEHMEN

Selector Name and related Constants	1 TR6 Reference
->NSF_CHANNEL	B-Kanalnummer, Octet 7
Basisanschluß:	
#B1_CHANNEL	*B1 Kanal*
#B2_CHANNEL	*B2 Kanal*
Primärmultiplexanschluß:	
wie "channel number" in "Channel Identification"	

6.1.7.4. PARAMETER FIELD FOR FAC. CODE #DIENSTWECHSEL_1 / #DIENSTWECHSEL_2

Selector Name and related Constants	1 TR6 Reference
->NSF_EAZ	EAZ der initiierenden Seite, Octet 7 * (length: 1 Byte)
->NSF_EAZ_ABH	EAZ der abhängigen Seite Octet 8 * (length: 1 Byte)

6.1.7.5. PARAMETER FIELD FOR FAC. CODE #GBG

Selector Name and related Constants	1 TR6 Reference
->NSF_GBG	GBG, Octet 7 * (length: 2 Bytes)

6.1.8. DISPLAY (I#DSP)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->DSP_DISPLAY	Octet 3 * , (max. length: 40 Bytes)

6.1.9. KEYPAD (I#KEY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->K_KEYPAD	Keypad, Octet 3 * (max. length: 32 Bytes)

6.1.10. ORGINATION ADDRESS (I#OAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->OAD_ADDR_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL	Address number type, Octet 3 *Unknown* *International Number* *National Number*
->OAD_ADDR_PLAN #UNKNOWN_PLAN #ISDN_PLAN	Address numbering plan, Octet 3 *Unknown* *ISDN Numbering PLan E. 164*
->OAD_ADDRESS	Address Octet 4 * (max. length 16 Bytes)

6.1.11. DESTINATION ADDRESS (I#DAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->DAD_ADDR_TYPE **#UNKNOWN** **#INTERNATIONAL** **#NATIONAL**	Address number type, Octet 3 *Unknown* *International Number* *National Number*
->DAD_ADDR_PLAN **#UNKNOWN_PLAN** **#ISDN_PLAN**	Address numbering plan, Octet 3 *Unknown* *ISDN Numbering PLan E. 164*
->DAD_ADDRESS	Address Octet 4 * (max. length 16 Bytes)

6.1.12. USER-USER INFORMATION (I#UUI)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->UUI_INFO	User-User Information, Octet 3 * (max. length 255 Bytes)

6.2. NATIONAL CS 6 IES (W-ELEMENTS)

6.2.1. SERVICE INDICATOR (I#SIN)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4.1, OCTET_4.2

Selector Name and related Constants	1 TR6 Refernce
->SIN_SERVICE	Service, Octet 3
#FERNSPRECHEN	*Fernsprechen*
->SIN_ADD_INFO	Additional Info, Octet 4
#ISDN_3.1KHZ	*ISDN-Fernsprechen 3,1 kHz*
#ANALOG	*Fernsprechen analog*
#ISDN_7KHZ	*ISDN-Fernsprechen 7 kHZ*
#A/B-DIENSTE	*A/B-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#FAX_GRUPPE_2	*Fax, Gruppe 2*
#FAX_GRUPPE_3	*Fax, Gruppe 3*
#MODEM_DATEN	*Daten über Modem*
#BTX_MODEM	*Btx über Btx_Modem*
#X.21-DIENSTE	*X.21-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#UC_4	*User Class 4*
#UC_5	*User Class 5*
#UC_6	*User Class 6*
#UC_19	*User Class 19*
#FAX_GRUPPE_4	*Telefax Gruppe 4*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BTX_64KBIT/S	*Btx 64 kBit/s*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#DUE_64KBIT/S	*Datenübertragung 64 kBit/s*
Add. info octet f. sync. Übertragung:	
->SIN_SYNC_PAT	Synchrone bit pattern - OCTET_4.1
#SYNC_PAT	*Synchrone Übertragung mit Bitratenadapt.*
->SIN_SYNC_L1_PROT	Layer 1 Protocol - OCTET_4.1
#V110/X30_SYNC	*V.110, X.30 (ECMA 102)*
#V120	*V.120*

->SIN_SYNC_USR_RATE	User Rate - OCTET_4.1
#1200BIT/S	*1200 bit/s*
#1200/75BIT/S	*1200/75 bit/2*
#75/1200BIT/S	*75/1200 bit/s*
#2400BIT/S	*2400 bit/s*
#4800BIT/S	*4800 bit/s*
#9600BIT/S	*9600 bit/s*
#14400BIT/S	*14400 bit/s*
#19200BIT/S	*19200 bit/s*
#48000BIT/S	*48000 bit/s*
#56000BIT/S	*56000 bit/s*
#IN_BAND	*in-band negotiation*
Add. info octet f. async. Übertragung:	
->SIN_ASYNC_L1_PROT	Layer 1 Protocol - OCTET_4.2
#V110/X30_ASYNC	*V.110, X.30 (ECMA102)*
->SIN_DATA_BITS	Number of Databits - OCTET_4.2
#8_DATA_BITS	*8 data bits*
#7_DATA_BITS	*7 data bits*
->SIN_STOP_BITS	Number of Stopbits - OCTET_4.2
#1_STOP_BIT	*1 stop bit*
#2_STOP_BITS	*2 stop bits*
->SIN_PARITY	Parity - OCTET_4.2
#NO_PARITY	*No parity*
#EVEN_PARITY	*Even Parity*
->SIN_ASYNC_USR_RATE	User Rate - OCTET_4.2
#1200BIT/S	*1200 bit/s*
#1200/75BIT/S	*1200/75 bit/2*
#75/1200BIT/S	*75/1200 bit/s*
#2400BIT/S	*2400 bit/s*
#4800BIT/S	*4800 bit/s*
#9600BIT/S	*9600 bit/s*
#14400BIT/S	*14400 bit/s*
#19200BIT/S	*19200 bit/s*
#X.25-DIENSTE	*X.25-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#UC_8	*User Class 8*
#UC_9	*User Class 9*
#UC_10	*User Class 10*
#UC_11	*User Class 11*
#UC_13	*User Class 13*
#19.2KBIT/S	*19,2 kbit/s*
#TELETEX_64KBIT/S	*Teletex 64*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#MIXED_MODE_1TR6	*Mixed Mode*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*

#FERNWIRKEN	*Fernwirken*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#GRAFIKTELEFON	*Grafiktelefon*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BILDSCHIRMTEXT	*Bildschirmtext*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BILDTELEFON	*Bildtelefon*
->SIN_ADD_INFO	Additional Information, Octet 4
#TON_3.1KHZ	*Ton 3,1 kHz*
#TON_7KHZ	*Ton 7 kHz*
#BILD	*Bild*

6.2.2. CHARGING INFO (I#CIF)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Refernece
->CIF_GEA #ANZAHL_EINHEITEN	Art der Gebühreninformation, Octet 3 *Anzahl der aufsummierten Gebühreninformat.*
->CIF_ANZAHL	IA5 codiert '0' -'9', Octet 4 * (max. length 12 Bytes)

6.2.3. DATE (I#DTE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Refernece
->DTE_DATE	Date: 'TT.MM.YY-HH:MM', Octet 3 * (length 14 Bytes)

6.2.4. FACILITY SELECT (I#FSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6

Selector Name and related Constants	1 TR6 Reference
->FSE_FAC_CODE	Facility Code, Octet 3 (coded as octet 4 of NSF)
->FSE_SERVICE	Service, Octet 4 (coded as octet 3 of Service Indicator)

6.2.5. STATUS OF FACILITIES (I#SOF)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 REFERNCE
->SOF_N_DM	Number of "aktive Dienstmerkmale"
->SOF_DM_01 ... **->SOF_DM_16** **#SPERRE_ABGEHEND** **#UMLEITUNG_STAENDIG** **#UMLEITUNG_FALLWEISE** **#GBG** **#GEBUEHREN_ANZEIGER** **#DM_IDENTIFIZIERUNG** **#DM_NOT_SELECTED**	Aktivierte Dienstmerkmale, Octet 3 *Sperre gegen abgehende Verbindung* *Anrufweiterschaltung I* *Anrufweiterschaltung II* *Anschluss gehört zu GBG* *Gebührenanzeiger eingerichtet* *Rufnummeridentifizierung böswilliger Anrufer eingerichtet* *Special code: unselected byte,DM-?? fields containing this pattern will not be coded into the IE*

6.2.6. STATUS DES GERUFENEN TLN (I#SDT)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->SDT_STATUS	Status, Octet 3
#KEINE_ANGABE	*keine Angabe*
#TLN_WIRD_GERUFEN	*Teilnehmer wird angerufen*

6.2.7. ADDITIONALTRANSMISSION ATTRIBUTES (I#ATA)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->ATA_FLAG	Flag, Octet 3
#INDICATION	*Indication*
#REQUEST	*Request*
->ATA_ATTRIBUTES	Attributes, Octet 3
#KEIN_SATELLIT	*Keine Satellitenverbindung*
#EIN_SATELLIT	*Eine Satellitenverbindung*
#ZWEI_SATELLIT	*Zwei Satellitenverbindungen*
#DREI_SATELLIT	*Drei Satellitenverbindungen*

ISDN LAYER 3 EUROPEAN VARIANT (ETSI)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2 Rev.1
Revision Date	:	May, 1992
Engineering	:	AUT E33 D / MNO
Quality Control	:	MWB KQP R.

C73000-M6076-C177 A02

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. Shift IE (I#SHIFT) 6-1

6.2. Congestion Level IE (I#CONG_LEVEL) 6-1

6.3. Segmented Message IE (I#SEG_MSG) 6-2

6.4. Bearer Capability IE (I#BEARER_CAP) 6-2

6.5. Cause IE (I#CAUSE) 6-7

6.6. Call Identity IE (I#CALL_ID) 6-9

6.7. Call State IE (I#CALL_STATE) 6-10

6.8. Channel Identification IE (I#CHANNEL_ID) 6-11

6.9. Facility IE (I#FACIL) 6-12

6.10. Progress Indicator IE (I#PROGRESS_IND) 6-15

6.11. Network-Specific Facility IE (I#NETWORK_FACIL) 6-16

6.12. Notification Indicator IE (I#NOTIFIC_IND) 6-16

6.13. Display IE (I#DISPLAY) 6-17

6.14. Date/Time IE (I#DATE) 6-17

6.15. Keypad IE (I#KEYPAD) 6-17

6.16. Signal IE (I#SIGNAL) 6-18

6.17. Calling Number IE (I#CALLING_NUM) 6-19

6.18. Calling Subaddress IE (I#CALLING_SAD) 6-20

6.19. Called Number IE (I#CALLED_NUM) 6-20

6.20. Called Subaddress IE (I#CALLED_SAD) 6-21

6.21. Redirecting Number IE (I#REDIRECTING_NUM) 6-21

6.22. Transit Network Selection IE (I#TRANS_NW_SEL) 6-22

6.23. Restart Indicator IE (I#RESTART_IND) 6-23

6.24. Low Layer Compatability IE (I#LOW_LAY_COMP) 6-24
6.25. High Layer Compatability IE (I#HI_LAY_COMP) 6-29
6.26. User User Information IE (I#UU_INFO) 6-30
6.27. Escape for Extension IE (I#ESCAPE) 6-30

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

Message Identifier	T/S 46-30 reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#CON_CON	Congestion Control
M#FAC	Facility
M#INFO	Info
M#NOTIFY	Notify
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry
M#SEGMENT	Segmented Message

3. INFORMATION ELEMENT TYPE IDENTIFIER

IE Identifier	T/S 46-30 reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#TRANS_NW_SEL	Transit Network Selection
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information
I#ESCAPE	Escape for Extension

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,	OCTET_3.1,	OCTET_3.2,
OCTET_3.3,				
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,	OCTET_6.1,	OCTET_6.2,	OCTET_6.3,
OCTET_6.4,	OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T301	#T301	180	180
Timer T302	#T302	15	10
Timer T303	#T303	4	4
Timer T304	#T304	30	20
Timer T305	#T305	30	30
Timer T306	#T306	30	30
Timer T307	#T307	180	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	40	35
Timer T312	#T312	6	6
Timer T313	#T313	4	4
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	60	60
Timer T318	#T318	4	4
Timer T319	#T319	4	4
Timer T320	#T320	30	30
Timer T321	#T321	30	30
Timer T322	#T322	4	4
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. Shift IE (I#SHIFT)

Selector Name and related Constants	T/S 46-30 Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	Codeset Specified in Shift IE
#CODESET0	*Q.931 IE*
#CODESET5	*CEPT defined IE*
#CODESET6	*National network specific IE*
#CODESET7	*User Specific IE*

6.2. Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	T/S 46-30 Reference
->CL_CONGESTION_LEVEL	Congestion Level
#RECEIVER_READY	*Receiver Ready*
#RECEIVER_NOT_READY	*Receiver not Ready*

6.3. Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.4. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_7

Selector Name and related Constants	T/S 46-30 Reference
->BC_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for present network*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*

Selector Name and related Constants	T/S 46-30 Reference
->BC_TRANSFER_RATE	Information Transfer Rate, Octet 4
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_STRUCTURE	Structure, Octet 4a
#DEFAULT	*Default*
#8KHZ_INTEGRITY	*8 kHz Integrity*
#SDU_INTEGRITY	*Service Data Unit Integrity*
#UNSTRUCTURED	*Unstructured*
->BC_CONFIGURATION	Configuration, Octet 4a
#POINT_TO_POINT	*Point-to-point*
->BC_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->BC_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
->BC_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->BC_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.7XX 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->BC_SYNC/ASYNC	Synchronous or Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*

Selector Name and related Constants	T/S 46-30 Reference
->BC_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->BC_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6 and X.1*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation V.6 and X.1*
#1.2/0.075KBIT/S	*1.2/0.075 kbit/s Recommendation V.6 and X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->BC_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*not used*
#INT_8KBIT/S	*8 kbit/s*
#INT_16KBIT/S	*16 kbit/s*
#INT_32KBIT/S	*32 kbit/s*
->BC_NIC_ON_TX	Network internal clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->BC_NIC_ON_RX	Network internal clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*

Selector Name and related Constants	T/S 46-30 Reference
->BC_FLOW_CTRL_TX #NOT_REQUIRED #REQUIRED	Flow Control on TX, Octet 5b.1 *No Flow Control required* *Flow Control required*
->BC_FLOW_CTRL_RX #NOT_ACCEPT #ACCEPT	Flow Control on RX, Octet 5b.1 *Cannot accept Flow Control* *Can accept Flow Control*
->BC_RATE_HEADER #NOT_INCLUDED #INCLUDED	Rate Adaption Header, Octet 5b.2 (Note 2) *Header not included* *Header included*
->BC_MULTI_FRAME #NOT_SUPPORTED #SUPPORTED	Multi Frame establishment, Octet 5b.2 *Not supported, only UI allowed* *Supported*
->BC_MODE #BIT_TRANSPARENT #PROTOCOL_SENSITIVE	Mode of operation, Octet 5b.2 *Bit transparent mode* *Protocol sensitive mode*
->BC_LLI_NEG #DEFAULT #FULL	Logical Link negotiation, Octet 5b.2 *Default, LLI=256 only* *Full protocol negotiation*
->BC_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assigner'* *Originator is 'Assigner only'*
->BC_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out-band negotiation, Octet 5b.2 *Negotiation is done 'Out-band'* *Negotiation is done 'In-band'*
->BC_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of stop bits, Octet 5c *Not used* *1 bit* *1.5 bits* *2 bits*
->BC_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of data bits, Octet 5c *Not used* *5 bits* *7 bits* *8 bits*

Selector Name and related Constants	T/S 46-30 Reference
->BC_PARITY	Parity information, Octet 5c
#ODD_PARITY	*odd*
#EVEN_PARITY	*even*
#NO_PARITY	*none*
#FORCED_TO_0	*forced to 0*
#FORCED_TO_1	*forced to 1*
->BC_DUPLEX_MODE	Duplex mode, Octet 5d
#HALF_DUPLEX	*Half duplex*
#FULL_DUPLEX	*Full duplex*
->BC_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
#V.35	*V.35*
->BC_LAYER2_ID	Layer Identification, Octet 6
#LAYER2	*User Info Layer 2 Protocol*
->BC_L2_PROTOCOL	Layer 2 Protocol Identifier, Octet 6
#Q.921	*Recommendation Q.921/I.441*
#X.25_LINK	*Recommendation X.25, Link Layer*
->BC_LAYER3_ID	Layer 3 Identification, Octet 7
#LAYER3	*User Info Layer 3 Protocol*
->BC_L3_PROTOCOL	Layer 3 Protocol Identifier, Octet 7
#Q.931	*Recommendation Q.921/I.441*
#X.25_PACKET	*Recommendation X.25, Packet Layer*

Note 1 : Octet 5b.1 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V120

6.5. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_5

Selector Name and related Constants	T/S 46-30 Reference
->C_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other international Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->C_LOCATION	Location, Octet 3
#USER	*User*
#LOACAL_PRIVATE	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#TRANSIT	*Transit Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->C_RECOMMENDATION	Recommendation, Octet 3a
#REC_Q.931	*Recommendation Q.931*
#REC_X.21	*Recommendation X.21*
#REC_X.25	*Recommendation X.25*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL0	*Normal Event Class 0*
->C_CAUSE_VALUE	Cause Value, Octet 4
#UNASSIGNED_NUMBER	*Unassigned Number*
#NO_ROUTE_TO_TRANSIT	*No Route to Transit Network*
#NO_ROUTE_TO_DEST	*No Route to Destination*
#CHANNEL_UNACCEPTABLE	*Channel Unacceptable*
#CALL_AWARDED	*Call awarded*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NON_SELECTED_CLEAR	*Non selected user clearing*

Selector Name and related Constants	T/S 46-30 Reference
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility Rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
->C_CAUSE_CLASS	Cause Class, Octet 4
#RESOURCE_UNAVAIL	*Resource not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NO_CHANNEL_AVAIL	*No Circuit/Channel Available*
#OUT_OF_ORDER	*Network Out of Order*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access Information Discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#QUALITY_UNAVAIL	*Quality of service unavailable*
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer Capability not Authorized*
#BEARER_UNAVAIL	*Bearer Capability not Available*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#CHANNEL_TYPE_UNIMPL	*Channel Type not Implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#ONLY_RESTRICTED	*Restricted only*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INVALID_MESSAGE	*Invalid Message*
->C_CAUSE_VALUE	Cause Value, Octet 4
#INVALID_CALL_REF	*Invalid Call Reference Value*
#CHANNEL_NONEXISTENT	*Channel not Existant*
#CALL_ID_NONEXISTENT	*Call Identity does not Exist*
#CALL_ID_IN_USE	*Call Identity in Use*
#NO_CALL_SUSPENDED	*No call suspended*
#CALL_CLEARED	*Call has been cleared*
#NONEXISTENT_CUG	*Non-existent closed user group*
#USER_NOT_MEMBER_CUG	*Called user not member of CUG*
#USER_BELONGS_CUG	*Called user belongs to CUG*

Selector Name and related Constants	T/S 46-30 Reference
#INV_TRANS_NET #INVALID_MESSAGE_UNSPEC	*Invalid transit network selection* *Invalid Message*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on Timer Expiry* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.6. Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.7. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->CS_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*
#CALL_ABORT	*Call Abort*
#OVERLAP_RECEIVING	*Overlap Receiving*

6.8. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_3.3

Selector Name and related Constants	T/S 46-30 Reference
->CID_INT_PRESENT #EXPLICIT #IMPLICIT	Interface Identifier Present, Octet 3 *Explicitly Identified* *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Other Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #D_CHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *D-Channel Indentified* *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_INT_ID	Interface identifier, Octet 3.1 *
->CID_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3.2 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 3.2 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS #H0_CHANNEL_UNITS #H11_CHANNEL_UNITS #H12_CHANNEL_UNITS	Channel or Map Type, Octet 3.2 *B-Channel Units* *H0-Channel Units* *H11-Channel Units* *H12-Channel Units*
->CID_NUMBER	Channel number, Octet 3.3

6.9. Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	T/S 46-30 Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	T/S 46-30 Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #USER_NOT_SUBSCRIBED #PROVIDED_BY_NETWORK	Error Value, Octet 6.5 (Note 3) *User not subscribed* *Provided by network*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	T/S 46-30 Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.10. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->PI_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3.2 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->PI_LOCATION #USER #LOCAL_PRIVAT #LOCAL_PUBLIC #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Remote Public Network* *Remote Private Network* *International network* *Network Beyond Interworking Point*
->PI_DESCRIPTION #NOT_END_TO_END #DEST_NON_ISDN #ORIG_NON_ISDN #RETURNED_TO_ISDN #INBAND_INFO_AVAIL	Progress Description, Octet 4 *Call is not end-to-end ISDN* *Destination Address is non-ISDN* *Origination Address is non-ISDN* *Call has Returned to the ISDN* *In-band Info now Available*

6.11. Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3.1 *User specified* *National network id.* *International network id.*
->NF_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3.1 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->NF_NET_ID	Network Identification, Octet 3.2 *
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.12. Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE #EXT_TO_COMPONENT	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change* *Extension to component usage*
->NI_COMPONENT	Component, Octet 4 *

6.13. Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->D_DISPLAY	Display, Octet 3 *

6.14. Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7

Selector Name and related Constants	T/S 46-30 Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7

6.15. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->K_KEYPAD	Keypad, Octet 3 *

6.16. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on - Pattern 0*
#ALERTING_ON_1	*Alerting on - Pattern 1*
#ALERTING_ON_2	*Alerting on - Pattern 2*
#ALERTING_ON_3	*Alerting on - Pattern 3*
#ALERTING_ON_4	*Alerting on - Pattern 4*
#ALERTING_ON_5	*Alerting on - Pattern 5*
#ALERTING_ON_6	*Alerting on - Pattern 6*
#ALERTING_ON_7	*Alerting on - Pattern 7*
#ALERTING_OFF	*Alerting off*

6.17. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLGN_NUMBER_TYPE	Type of Number, Octet 3
#UNKNOWN	*Unknown*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
#SUBSCRIBER	*Subscriber Number*
#ABBREVIATED	*Abbreviated Number*
->CLGN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#UNKNOWN_PLAN	*Unknown*
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#DATA_PLAN	*Data Numbering Plan X.121*
#TELEX_PLAN	*Telex Numbering Plan F.69*
#NATIONAL_PLAN	*National Numbering Plan*
#PRIVATE_PLAN	*Private Numbering Plan*
->CLGN_PRESENTATION	Presentation Indicator, Octet 3
#PRESENT_ALLOWED	*Presentation Allowed*
#PRESENT_RESTRICTED	*Presentation Restricted*
#NUMBER_UNAVAIL	*Number Not Available Due to Interworking*
->CLGN_SCREENING	Screening indicator, Octet 3a
#UNSCREENED	*User-Provided, Not Screened*
#VERIFY_PASSED	*User-Provided, Verified and Passed*
#VERIFY_FAILED	*User-Provided, Verified and Failed*
#NETWORK_PROVIDED	*Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.18. Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.19. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLDN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber Number* *Abbreviated Number*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.20. Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLDS_ADDRESS_TYPE **#USER_SPECIFIED** **#NSAP**	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLDS_ODD/EVEN **#EVEN_NUMBER** **#ODD_NUMBER**	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.21. Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->RDGN_NUMBER_TYPE **#UNKNOWN** **#INTERNATIONAL** **#NATIONAL** **#NETWORK_SPECIFIC** **#SUBSCRIBER** **#ABBREVIATED**	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber number* *Abbreviated Number*
->RDGN_NUMBERING_PLAN **#UNKNOWN_PLAN** **#ISDN_PLAN** **#DATA_PLAN** **#TELEX_PLAN** **#NATIONAL_PLAN** **#PRIVATE_PLAN**	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*

Selector Name and related Constants	T/S 46-30 Reference
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->RDGN_REASON #CALL_FORW_BUSY #CALL_FORW_NO_REPLY #CALLED_OUT_OF_ORDER #FORW_BY_CALLED #UNCOND_OR_SYS	Reason for redirection, Octet 3b *Call forwarding or called DTE busy* *Call forwarding no reply* *Called DTE out of order* *Forwarding by called DTE* *Unconditional or systematic red.*
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.22. Transit Network Selection IE (I#TRANS_NW_SEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->TR_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3 *User specified* *National network id.* *International network id.*
->TR_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->TR_NET_ID	Network Identification, Octet 4 *

6.23. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->RI_CLASS	Class of Channel or Interface, Octet 3
#INDICATED_CHANNEL	*Indicated Channel*
#SINGLE_INTERFACE	*Single Interface*
#ALL_INTERFACES	*All Interface*

6.24. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	T/S 46-30 Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 2.1 or 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation indicator, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S #2x64KBIT/S #384KBIT/S #1536KBIT/S #1920KBIT/S	Information Transfer Rate, Octet 4 *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*

Selector Name and related Constants	T/S 46-30 Reference
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC	Symmetry, Octet 4b *Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B #CHANNEL_SIZE #64KBIT/S #2x64KBIT/S #384KBIT/S #1536KBIT/S #1920KBIT/S	Information Transfer Rate (D->O), Octet 4b *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->LL_L1_PROTOCOL #RATE_ADAPTION_V110/X30 #G.711_ULAW #G.711_ALAW #G.721_ADPCM #G.7XX_AUDIO #G.7XX_VIDEO #NON_CCITT #RATE_ADAPTION_V120 #RATE_ADAPTION_X31	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption V.110/X.30* *Recommendation G.711 u-law* *Recommendation G.711 A-law* *Recommendation G.721 32 kbit/s ADPCM* *Recommendation G.7XX 7kHz Audio* *Recommendation G.711 384 kbit/s Video* *Non-CCITT rate adaption* *Rate Adaption V.120* *Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC #SYNCHRONOUS #ASYNCHRONOUS	Synchronous/Asynchronous, Octet 5a *Synchronous* *Asynchronous*
->LL_NEGOTIATION #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation, Octet 5a *In-band negotiation not possible* *In-band negotiation possible*
->LL_USER_RATE #E_BITS #0.6KBIT/S #1.2KBIT/S #2.4KBIT/S #3.6KBIT/S #4.8KBIT/S #7.2KBIT/S #8KBIT/S #9.6KBIT/S #14.4KBIT/S #16KBIT/S #19.2KBIT/S	User rate, Octet 5a *Indicated by E-bits, Recommendation I.460* *0.6kbit/s Recommendation V.6 and X.1* *1.2kbit/s Recommendation V.6* *2.4kbit/s Recommendation V.6* *3.6kbit/s Recommendation V.6* *4.8kbit/s Recommendation V.6 and X.1* *7.2kbit/s Recommendation V.6* *8kbit/s Recommendation I.460* *9.6kbit/s Recommendation V.6 and X.1* *14.4kbit/s Recommendation V.6* *16kbit/s Recommendation I.460* *19.2kbit/s Recommendation V.6*

Selector Name and related Constants	T/S 46-30 Reference
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*Not Used*
#INT_8KBIT/S	*8kbit/s*
#INT_16KBIT/S	*16kbit/s*
#INT_32KBIT/S	*32kbit/s*
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*Flow Control not required*
#REQUIRED	*Flow Control Required*
->LL_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot Accept Flow Control*
#ACCEPT	*Can Accept Flow Control*
->LL_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->LL_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->LL_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*

Selector Name and related Constants	T/S 46-30 Reference
->LL_LLI_NEG	Logical Link negotiation, Octet 5b.2
#DEFAULT	*Default, LLI=256 only*
#FULL	*Full protocol negotiation*
->LL_ASSIGNER	Assignor/Assignee, Octet 5b.2
#DEFAULT	*Originator is 'Default assigner'*
#ONLY_ASSIGNER	*Originator is 'Assigner only'*
->LL_IN/OUT_BAND	In/Out-band negotiation, Octet 5b.2
#OUT_BAND	*Negotiation is done 'Out-band'*
#IN_BAND	*Negotiation is done 'In-band'*
->LL_STOP_BITS	Number of Stop Bits, Octet 5c
#NOT_USED	*Not used*
#1_STOP_BIT	*1 Bit*
#1.5_STOP_BITS	*1.5 Bits*
#2_STOP_BITS	*2 Bits*
->LL_DATA_BITS	Number of Data Bits, Octet 5c
#5_DATA_BITS	*5 Bits*
#7_DATA_BITS	*7 Bits*
#8_DATA_BITS	*8 Bits*
->LL_PARITY	Parity Information, Octet 5c
#ODD_PARITY	*Odd*
#EVEN_PARITY	*Even*
#NO_PARITY	*None*
#FORCED_TO_0	*Forced to 0*
#FORCED_TO_1	*Forced to 1*
->LL_DUPLEX_MODE	Duplex mode, Octet 5d
#HALF_DUPLEX	*Half duplex*
#FULL_DUPLEX	*Full duplex*
->LL_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
#V.35	*V.35*

Selector Name and related Constants	T/S 46-30 Reference
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #ISO_1745 #Q.921 #X.25_LINK #X.25_MULTILINK #EXTENDED_LAPB #HDLC_ARM #HDLC_NRM #HDLC_ABM #LAN_LLC #X.75	Layer 2 Protocol Identifier, Octet 6 *Basic mode ISO 1745* *Recommendation Q.921 (I.441)* *Recommendation X.25, Link level* *Recommendation X.25 Multilink* *Extended LAPB; half duplex (T.71)* *HDLC ARM (ISO 4335)* *HDLC NRM (ISO 4335)* *HDLC ABM (ISO 4335)* *LAN Logical link control (ISO 8802/2)* *Rec. X.75, single link procedure*
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET #ISO_8208 #ISO_8348 #ISO_8473 #T.70	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer* *ISO 8208* *ISO 8348* *ISO 8473* *Rec. T.70, minimum network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.25. High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A

Selector Name and related Constants	T/S 46-30 Reference
->HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->HL_INTERPRETATION	Interpretation, Octet 3
#FIRST_CHARACTER	*First HL Characteristics to be Used*
->HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
->HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#MAINTENANCE	*Reserved for Maintenance (Note 1)*
#MANAGEMENT	*Reserved for Management (Note 1)*
->HL_CHARACTER_EXT	Extended High Layer Characteristics, Octet 4a
#TELEPHONY	*Telephony*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application*
#NOT_AVAIL1	*Not available for assignment*
#NOT_AVAIL2	*Not available for assignment*

Note 1 : Only in this case, Octet 4a may follow

6.26. User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->UU_PROTOCOL_DISCR	Protocol discriminator, Octet 3
#USER_SPECIFIC_PD	*User specific protocol*
#OSI_HIGHER_LAYER	*OSI Higher Layer Protocols*
#X.244	*X.244*
#SYS_MNG_CONVERG	*System mng. convergence function*
#IA5_CHAR	*IA5 characters*
#V.120	*reserved for V.120 rate adaption*
#Q.931_CALL_CONTROL	*Q931 Call Control Messages*
->UU_USER_INFO	User-User Information, Octet 4 *

6.27. Escape for Extension IE (I#ESCAPE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->EX_IE_ID	Information element identifier, Octet 3
->EX_ESCAPE	Contents of information element, Octet 4 *

ISDN LAYER 3 NATIONAL VARIANT (AT&T : 5ESS)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version : V2.0
System Version : MF 2.1

Revision Date : August, 1990

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

2.1. ONE-OCTET MESSAGES 2-1

2.2. TWO-OCTET MESSAGES 2-2

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

3.1. CODESET 0 IES 3-1

3.2. CODESET 6 IES 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. CODESET 0 IES 6-1
6.1.1. Shift IE (I#SHIFT) 6-1
6.1.2. Bearer Capability IE (I#BEARER_CAP) 6-2
6.1.3. Cause IE (I#CAUSE) 6-3
6.1.4. Call State IE (I#CALL_STATE) 6-5
6.1.5. Channel Identification IE (I#CHANNEL_ID) 6-5
6.1.6. Progress Indicator IE (I#PROGRESS_IND) 6-6
6.1.7. Terminal Capabilities IE (I#TERM_CAPAB) 6-6
6.1.8. Keypad IE (I#KEYPAD) 6-6
6.1.9. Signal IE (I#SIGNAL) 6-7
6.1.10. Switchhook IE (I#SWITCHHOOK) 6-7
6.1.11. Calling Number IE (I#CALLING_NUM) 6-8
6.1.12. Called Number IE (I#CALLED_NUM) 6-8
6.1.13. Restart Indicator IE (I#RESTART_IND) 6-9
6.1.14. Low Layer Compatability IE (I#LOW_LAY_COMP) 6-10

6.2. CODESET 6 IES 6-12
6.2.1. User Code IE (I#USER_CODE) 6-12
6.2.2. Associated Type IE (I#ASSOC_TYPE) 6-12
6.2.3. Selected Call Appearance IE (I#SELECT_CALL) 6-12
6.2.4. Origination Call Appearance IE (I#ORIG_CALL) 6-13
6.2.5. Destination Call Appearance IE (I#DEST_CALL) 6-13
6.2.6. Endpoint Identifier IE (I#ENDPOINT_ID) 6-13
6.2.7. Keypad Control IE (I#KEYPAD_CTRL) 6-13
6.2.8. Other Call Reference IE (I#OTHER_CR) 6-14
6.2.9. Feature Activation IE (I#FEATURE_ACT) 6-14

6.2.10. Feature Indication IE (I#FEATURE_IND) 6-15
6.2.10.1. Further Octets for Status Type: #FEATURE_NUMBER_STATUS 6-15
6.2.10.2. Further Octets for Status Type: #FEATURE_BUTTON_STATUS 6-16
6.2.10.3. Further Octets for Status Type: #MULTIPLE_BUTTON_STATUS 6-16
6.2.10.4. Further Octets for Status Type: #MAINTENANCE_STATUS ... 6-16
6.2.10.5. Further Octets for Status Type: #MAINTENANCE_STATUS ... 6-16
6.2.11. Adjunct Control IE (I#ADJUNCT_CTRL) 6-17
6.2.12. Display Control IE (I#DISPLAY_CTRL) 6-17
6.2.13. Display Field IE (I#DISPLAY_FLD) 6-18
6.2.14. Management IE (I#MANAGEMENT) 6-19
6.2.14.1. Parameter Field for Parameter Identifier: #SERVICE_PROFILE_ID 6-20
6.2.14.2. Parameter Field for Parameter Identifier: #ENDPOINT_ID 6-20
6.2.14.3. Parameter Field for Parameter Identifier: #CALL_REF_ID 6-20
6.2.14.4. Parameter Field for Parameter Identifier: #CALL_STATE_ID .. 6-21
6.2.14.5. Parameter Field for Parameter Identifier: #CHANNEL_ID 6-21
6.2.14.6. Parameter Field for Parameter Identifier: #SERVICE_STATE .. 6-21
6.2.14.7. Parameter Field for Parameter Identifier: #SERVICE_MSG_ERROR 6-22
6.2.14.8. Parameter Field for Parameter Identifier: #LOOPBACK_LOCATION 6-22
6.2.14.9. Parameter Field for Parameter Identifier: #SUPPL_CAPABILITIES 6-22
6.2.14.10. Parameter Field for Parameter Identifier: #CAUSE_ID 6-23
6.2.14.11. Parameter Field for Other Parameter Identifier 6-23

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

2.1. ONE-OCTET MESSAGES

Message Identifier	AT&T Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#INFO	Info
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

2.2. TWO-OCTET MESSAGES

Message Identifier	AT&T Reference
M#CONF	Conference
M#CONF_ACK	Conferene Acknowledge
M#CONF_REJ	Conference Reject
M#TRANS	Transfer
M#TRANS_ACK	Transfer Acknowledge
M#TRANS_REJ	Transfer Reject
M#ASSOC	Associated
M#ASSOC_ACK	Associated Acknowledge
M#DROP	Drop
M#DROP_ACK	Drop Acknowledge
M#DROP_REJ	Drop Reject
M#HOLD	Hold
M#HOLD_ACK	Hold Acknowledge
M#HOLD_REJ	Hold Reject
M#RECONN	Reconnect
M#RECONN_ACK	Reconnect Acknowledge
M#RECONN_REJ	Reconnect Reject
M#REDIR	Redirect
M#MAN_INFO	Management Information

3. INFORMATION ELEMENT TYPE IDENTIFIER

3.1. CODESET 0 IES

IE Identifier	AT&T Reference
I#SHIFT	Codeset Shift
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#PROGRESS_IND	Progress Indicator
I#TERM_CAPAB	Terminal Capabilities
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#SWITCHHOOK	Switchhook
I#CALLING_NUM	Calling Party Number
I#CALLED_NUM	Called Party Number
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability

3.2. CODESET 6 IES

IE Identifier	AT&T Reference
I#USER_CODE	User Code
I#ASSOC_TYPE	Associated Type
I#SELECT_CALL	Selected Call Appearance
I#ORIG_CALL	Origination Call Appearance
I#DEST_CALL	Destination Appearance
I#ENDPOINT_ID	Endpoint Identidier
I#KEYPAD_CTRL	Keypad Control
I#OTHER_CR	Other Call Reference
I#FEATURE_ACT	Feature Activation
I#FEATURE_IND	Feature Indication
I#ADJUNCT_CTRL	Adjunct Control
I#DISPLAY_CTRL	Display Control
I#DISPLAY_FLD	Display Field
I#MANAGEMENT	Management

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,			
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,			
OCTET_6,				
OCTET_7,	OCTET_7C,	OCTET_7D,	OCTET_7E,	OCTET_7F,
OCTET_7G,	OCTET_7H,	OCTET_7I,	OCTET_7I.1,	OCTET_7I.2,

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)
Timer T302	#T302	10
Timer T303	#T303	4
Timer T305	#T305	4
Timer T308	#T308	4
Timer T310	#T310	4
General Purpose Timer 1	#USER1	10
General Purpose Timer 2	#USER2	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. CODESET 0 IES

6.1.1. Shift IE (I#SHIFT)

Selector Name and related Constants	AT&T Reference
->SH_TYPE #NON_LOCKING #LOCKING	Locking/Non-locking Shift Type *Non Locking* *Locking*
->SH_CODESET #CODESET0 #CODESET5 #CODESET6 #CODESET7	Codeset Specified in Shift IE *Q.931 IE* *National Use IE* *Local Network Specific IE* *User Specific IE*

1.1.2. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_5A, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->BC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate, Octet 4 *Used for packet mode* *64 kbit/s*
->BC_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->BC_L1_PROTOCOL #RATE_ADAPTION_V110/X30 #G.711_ULAW	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption V.110/X.30* *Recommendation G.711 u-law*
->BC_SYNC/ASYNC #SYNCHRONOUS	Synchronous or Asynchronous, Octet 5a *Synchronous*
->BC_USER_RATE #56KBIT/S	User rate, Octet 5a *56kbit/s* *Recommendation V.6*
->BC_LAYER2_ID #LAYER2	Layer Identification, Octet 6 *User Info Layer 2 Protocol*
->BC_L2_PROTOCOL #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation X.25, Link Layer*
->BC_LAYER3_ID #LAYER3	Layer 3 Identification, Octet 7 *User Info Layer 3 Protocol*
->BC_L3_PROTOCOL #X.25_PACKET	Layer 3 Protocol Identifier, Octet 7 *Recommendation X.25, Packet Layer*

6.1.3. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4,

Selector Name and related Constants	AT&T Reference
->C_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->C_LOCATION #USER #LOACAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Transit Network* *Remote Public Network* *Remote Private Network* *International Network* *Network Beyond Interworking Point*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER #NO_ROUTE_TO_TRANSIT	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned Number* *No Route to Transit Network*
->C_CAUSE_CLASS #NORMAL1 ->C_CAUSE_VALUE #NORMAL_CLEARING #USER_BUSY #NO_USER_RESPOND #CALL_REJECTED #NUMBER_CHANGED #INVALID_NUMBER_FORMAT #FACILITY_REJECTED #STATUS_ENQ_RESPONSE #NORMAL_UNSPECIFIED	Cause Class, Octet 4 *Normal Event Class 1* Cause Value, Octet 4 *Normal call clearing* *User Busy* *No User Responding* *Call Rejected* *Number Changed* *Invalid Number Format* *Facility Rejected* *Response to STATUS ENQUIRY* *Normal, Unspecified*

Selector Name and related Constants	AT&T Reference
->C_CAUSE_CLASS #RESOURCE_UNAVAIL ->C_CAUSE_VALUE #NO_CHANNEL_AVAIL #QUEUED #TEMPORARY_FAILURE #SWITCH_CONGESTION #ACCESS_INFO_DISCARD	Cause Class, Octet 4 *Resource not Available* Cause Value, Octet 4 *No Circuit/Channel Available* *Queued* *Temporary Failure* *Switching Equipment Congestion* *Access Information Discarded*
->C_CAUSE_CLASS #SERVICE_UNAVAIL ->C_CAUSE_VALUE #FAC_NOT_SUBSCRIBED #OUTGOING_BARRED #INCOMING_BARRED #BEARER_UNAVAIL #SERVICE_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Service not Available* Cause Value, Octet 4 *Requested facility not subscribed* *Outgoing calls barred* *Incoming calls barred* *Bearer Capability not Available* *Service not Available*
->C_CAUSE_CLASS #SERVICE_UNIMPL ->C_CAUSE_VALUE #BEARER_SERVICE_UNIMPL #CHANNEL_TYPE_UNIMPL #FAC_UNIMPL	Cause Class, Octet 4 *Service not implemented* Cause Value, Octet 4 *Bearer capability not implemented* *Channel Type not Implemented* *Requested facility not implemented*
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTANT #INCOMPATIBLE_DEST #INV_TRANS_NET	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Incompatible Destination* *Invalid transit network selection*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #INVALID_IE_CONTENTS #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *Invalid IE Contents* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Interworking*

6.1.4. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#RELEASE_REQUEST	*Release Request*

6.1.5. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->CID_INT_PRESENT	Interface Identifier Present, Octet 3
#IMPLICIT	*Implicitly Identified*
->CID_INT_TYPE	Interface Type, Octet 3
#BASIC_INTERFACE	*Basic Interface*
->CID_PREF/EXCL	Preferred/Exclusive, Octet 3
#EXCLUSIVE	*Exclusive*
#PREFERRED	*Preferred*
->CID_DCHANNEL	D-channel Indicator, Octet 3
#D_CHANNEL	*D-Channel Indentified*
#NOT_D_CHANNEL	*Not D-Channel*
->CID_INFO_CHAN_SEL	Information Channel Selection, Octet 3
#NO_CHANNEL	*No Channel*
#B1_CHANNEL	*B1 Channel*
#B2_CHANNEL	*B2 Channel*
#ANY_CHANNEL	*Any Channel*

1.1.6. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->PI_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->PI_LOCATION #USER #LOCAL_PRIVAT #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #REMOTE_PRIVATE #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Transit Network* *Remote Public Network* *Remote Private Network* *Network Beyond Interworking Point*
->PI_DESCRIPTION #NOT_END_TO_END #DEST_NON_ISDN #ORIG_NON_ISDN #RETURNED_TO_ISDN #INBAND_INFO_AVAIL	Progress Description, Octet 4 *Call is not end-to-end ISDN* *Destination Address is non-ISDN* *Origination Address is non-ISDN* *Call has Returned to the ISDN* *In-band Info now Available*

1.1.7. Terminal Capabilities IE (I#TERM_CAPAB)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->TC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->TC_DESCRIPTION #TYPE2_STIMULUS	Capability Description, Octet 3 *Type 2 stimulus mode terminal*

1.1.8. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.9. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#CUSTOM_ON	*Custom tone on*
#RECALL_DIAL_ON	*Recall dial tone on*
#BUSY_VERIFY_ON	*Busy verify tone on*
#ERROR_ON	*Error tone on*
#STUTTER_DIAL_ON	*Stutter dial tone on*
#EXPENSIVE_ROUTING_ON	*Expensive routing tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*
#UNSPEC_TONE	*Unspecified tone*

6.1.10. Switchhook IE (I#SWITCHHOOK)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SW_VALUE	Switchhook Value, Octet 3
#ON_HOOK	*On-hook*
#OFF_HOOK	*Off-hook*

1.1.11. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	AT&T Reference
->CLGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Subscriber Number* *Abbreviated Number*
->CLGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED #NUMBER_UNAVAIL	Presentation Indicator, Octet 3a *Presentation Allowed* *Presentation Restricted* *Number Not Available Due to Interworking*
->CLGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

1.1.12. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->CLDN_NUMBER_TYPE #INTERNATIONAL #NATIONAL #LOCAL_DIRECTORY	Type of Number, Octet 3 *Other International Standards* *National Standard* *Local Directory Number*
->CLDN_NUMBERING_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.13. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->RI_CLASS #INDICATED_CHANNEL #SINGLE_INTERFACE	Class of Channel or Interface, Octet 3 *Indicated B-Channel* *Single Interface*

6.1.14. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate, Octet 4 *Used for packet mode* *64 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC #BIDIRECT_ASYMMETRIC #UNIDIRECT_ORIG_DEST #UNIDIRECT_DEST_ORIG	Symmetry, Octet 4b *Bidirectional Symmetric* *Bidirectional Asymmetric* *Unidirectional orig -> dest* *Unidirectional dest -> orig*

Selector Name and related Constants	AT&T Reference
->LL_TRANSFER_RATE_4B #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate (D->O), Octet 4b *Used for packet mode* *64 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->LL_L1_PROTOCOL #RATE_ADAPTION_V110/X30 #G.711_ULAW #G.711_ALAW #G.721_ADPCM #NON_CCITT	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption V.110 / X.30* *Recommendation G.711 u-law* *Recommendation G.711 A-law* *Recommendation G.721 32 kbit/s ADPCM* *Non-CCITT rate adaption*
->LL_SYNC/ASYNC #SYNCHRONOUS #ASYNCHRONOUS	Synchronous/Asynchronous, Octet 5a *Synchronous* *Asynchronous*
->LL_USER_RATE #UNDEFINED #0.6KBIT/S #1.2KBIT/S #2.4KBIT/S #3.6KBIT/S #4.8KBIT/S #7.2KBIT/S #8KBIT/S #9.6KBIT/S #14.4KBIT/S #16KBIT/S #19.2KBIT/S #32KBIT/S #48KBIT/S #56KBIT/S	User rate, Octet 5a *Undefined* *0.6kbit/s Recommendation V.6 and X.1* *1.2kbit/s Recommendation V.6* *2.4kbit/s Recommendation V.6* *3.6kbit/s Recommendation V.6* *4.8kbit/s Recommendation V.6 and X.1* *7.2kbit/s Recommendation V.6* *8kbit/s Recommendation I.460* *9.6kbit/s Recommendation V.6 and X.1* *14.4kbit/s Recommendation V.6* *16kbit/s Recommendation I.460* *19.2kbit/s Recommendation V.6* *32kbit/s Recommendation I.460* *48kbit/s Recommendation V.6 and X.1* *56kbit/s Recommendation V.6*
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #Q.921 #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation Q.921 (I.441)* *Recommendation X.25, Link level*
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer*

1.2. CODESET 6 IES

1.2.1. User Code IE (I#USER_CODE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->UC_TYPE	Type of User Code, Octet 3
#ANY_CODE	*Any user code*
#ACCOUNT_CODE	*Account code*
#LOGIN_DIGITS	*Login digits*
#SUBSCRIBER_ID	*Subsciber id*
#AUTHOR_CODE	*Authorization code*
->UC_VALUE	User Code Value, Octet 4 *

1.2.2. Associated Type IE (I#ASSOC_TYPE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->AT_STATUS_VALUE	Status Value, Octet 3
#SETUP	*Setup*
#CONNECT	*Connect*
#HOLD	*Hold*
#RECONNECT	*Reconnect*
#EXCLUSION	*Exclusion*
#CONNECT_DENIED	*Connect denied*
#CLEAR_DENIED	*Clearing denied*

1.2.3. Selected Call Appearance IE (I#SELECT_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SCA_BUTTON_NUM	Button Number, Octet 3

6.2.4. Origination Call Appearance IE (I#ORIG_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->OCA_VALUE	Origination Call Appearance Value, Octet 3

6.2.5. Destination Call Appearance IE (I#DEST_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->DCA_VALUE	Destination Call Appearance Value, Octet 3

6.2.6. Endpoint Identifier IE (I#ENDPOINT_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->EID_USER_SERVICE_ID	User Service Identifier, Octet 3
->EID_INTERPRETER	Interpreter, Octet 4
->EID_TERMINAL_ID	Terminal Identifier, Octet 4

6.2.7. Keypad Control IE (I#KEYPAD_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->KC_CR_VALUE_TYPE #NULL_CALL_REF #NON_NULL_CALL_REF	Call Reference Value Type, Octet 3 *Null call reference* *Non-null call reference*

6.2.8. Other Call Reference IE (I#OTHER_CR)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->OCR_FLAG #ORIG_SIDE #DEST_SIDE	Flag, Octet 3 *Origination side* *Destination side*
->OCR_VALUE	Other Call Reference Value, Octet 3

6.2.9. Feature Activation IE (I#FEATURE_ACT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->FA_BUTTON_TYPE #UNKNOWN #CALL_APPEARANCE #FEATURE_BUTTON	Button Type, Octet 3 *Unknown* *Call appearance button* *Feature button*
->FA_MODULE_NUM	Module Number, Octet 3
->FA_STATUS_TYPE	Status Type, Octet 3
->FA_BUTTON_NUM	Button Number, Octet 4

6.2.10. Feature Indication IE (I#FEATURE_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->FI_BUTTON_TYPE #UNKNOWN #CALL_APPEARANCE #FEATURE_BUTTON #NONE_APPLICABLE	Button Type, Octet 3 *Unknown* *Call appearance button* *Feature button* *None applicable*
->FI_MODULE_NUM	Module Number, Octet 3
->FI_STATUS_TYPE #FEATURE_NUMBER_STATUS #FEATURE_BUTTON_STATUS #MULTILE_BUTTON_STATUS #MAINTENANCE_STATUS	Status Type, Octet 3 *Feature number status* *Feature button status* *Multiple button status* *Maintenance status*

6.2.10.1. Further Octets for Status Type: #FEATURE_NUMBER_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Number, Octet 4
->FI_STATUS #ACTIVATED #DEACTIVATED #PENDING #LOCAL_HOLD #REMOTE_HOLD #CONFIRMED #REQUESTED_STATE #REJECTED	Status, Octet 5 *Activated* *Deactivated* *Pending* *Local hold* *Remote hold* *Confirmed* *Already in requested state* *Rejected*
->FI_ADD_INFO	Additional Information, Octet 6 * (for repetition of Octet 4 and Octet 5)

1.2.10.2. Further Octets for Status Type: #FEATURE_BUTTON_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Number, Octet 4
->FI_INDICATOR	Indicator, Octet 5
->FI_STATUS #ACTIVATED #DEACTIVATED #PENDING #LOCAL_HOLD #REMOTE_HOLD #CONFIRMED #REQUESTED_STATE #REJECTED	Status, Octet 5 *Activated* *Deactivated* *Pending* *Local hold* *Remote hold* *Confirmed* *Already in requested state* *Rejected*
->FI_ADD_INFO	Additional Information, Octet 6 * (for repetition of Octet 4 and Octet 5)

1.2.10.3. Further Octets for Status Type: #MULTIPLE_BUTTON_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Button Number, Octet 4
->FI_NUM_BUTTONS	Number of Buttons, Octet 5
->FI_STATUS	Status, Octet 6
->FI_ADD_INFO	Additional Information, Octet 7 * (for more Status Information)

1.2.10.4. Further Octets for Status Type: #MAINTENANCE_STATUS

Selector Name and related Constants	AT&T Reference
->FI_MAINTENANCE #ALL_INDICATORS_OFF	Maintenance, Octet 4 *Turn all indicators off*

1.2.10.5. Further Octets for other Status Type

Selector Name and related Constants	AT&T Reference
->FI_ADD_INFO	Additional Information, Octet 4 *

6.2.11. Adjunct Control IE (I#ADJUNCT_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->AC_VALUE	Adjunct Control Value, Octet 3
#OFF	*Off*
#ON	*On*

6.2.12. Display Control IE (I#DISPLAY_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->DC_MODE	Display mode, Octet 3
#NO_CHANGE	*No change*
#NORMAL	*Normal*
#INSPECT	*Inspect*
#MISC_DISPLAY_INFO	*Misc. display information*
#MSG_RETRIEVAL	*Message retrieval*
#ELECTRONIC_DIRECTORY	*Electronic directory query*

6.2.13. Display Field IE (I#DISPLAY_FLD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	AT&T Reference
->DF_MODE	Display Mode, Octet 3
#NORMAL	*Normal*
#INSPECT	*Inspect*
#MISC_DISPLAY_INFO	*Misc. display information*
#MSG_RETRIEVAL	*Message retrieval*
#ELECTRONIC_DIRECTORY	*Electronic directory query*
->DF_SUBMODE	Submode, Octet 3
#NO_SUBMODE	*No submode applicable*
#DIRECT	*Direct*
#REDIRECTED	*Redirected*
->DF_TYPE	Display Field Type, Octet 4
#CALL_APPEAR_ID	*Call appearance ID*
#CALLED_PARTY_ID	*Called party ID*
#CALLING_PARTY_ID	*Calling party ID*
#CALLED_PARTY_NAME	*Called party name*
#CALLING_PARTY_NAME	*Calling party name*
#ORIG_PERMISSION	*Originating permission*
#ISDN_CALL_ID	*ISDN call identification*
#MISC_CALL_INFO	*Misc. call information*
#ENTIRE_DISPLAY	*Entire display*
#DATE_AND_TIME	*Date and time of day*
->DF_INFO	Display Information, Octet 5 *

6.2.14. Management IE (I#MANAGEMENT)

Possible octet inclusions/exclusions:

OCTET_7C, OCTET_7D, OCTET_7E, OCTET_7F, OCTET_7G,
OCTET_7H, OCTET_7I, OCTET_7I.1, OCTET_7I.2

Selector Name and related Constants	AT&T Reference
->M_PROTOCOL_DISCR #AT&T_ISDN_PROTOCOL	Protocol Discriminator, Octet 7c *AT&T ISDN management protocol*
->M_TRANSACTION_REF	Transaction Reference, Octet 7d
->M_OP_CLASS #UNCONFIRMED_OPERATION #CONFIRMED_OPERATION #RETURN_RESULT #RETURN_RESULT_CONT #RETURN_ERROR #REJECT	Operation Class, Octet 7e *Unconfirmed operation* *Confirmed operation* *Return result* *Return result continued* *Return error* *Reject*
->M_OP_TYPE #ACTION #EVENT_REPORT #GET #SET	Operation Type, Octet 7e *Action* *Event report* *Get* *Set*
->M_CODE for Operation Type #ACTION #ACTIVATE_LOOPBACK #DEACTIVATE_LOOPBACK #RESET for Operation Type #EVENT_REPORT #INIT_REQUEST #ENDPOINT_SERV_CHANGE for Operation Type #GET #SERVICE_PROFILE_INFO #CALL_STATUS_INFO #EQUIPMENT_INFO for Operation Type #SET #SERVICE_PROFILE_INFO #ADDRESS_INFO for Operation Class #REJECT #UNSPEC_ERROR #PROTOCOL_VIOLATION #UNREC_OPERATION #NON_SYNC_OPERATION	Code, Octet 7f *Activate loopback* *Deactivate loopback* *Reset* *Initialization request* *Endpoint service state change* *Service profile information* *Call status information* *Equipment information* *Service profile information* *Address information* *Unspecified error* *Protocol violation* *Unrecognized operation* *Non-synchronous operation not supported*

Selector Name and related Constants	AT&T Reference
->M_PARAMETER_ID #SERVICE_PROFILE_ID #ENDPOINT_ID #CALL_REF_ID #CALL_STATE_ID #CHANNEL_ID #SERVICE_STATE #SERVICE_MSG_ERROR #LOOPBACK_LOCATION #SUPPL_CAPABILITIES #CAUSE_ID	Parameter Identifier, Octet 7g *Service profile identifier* *Endpoint identifier* *Call reference identifier* *Call state identifier* *Channel identifier* *Service state* *Service message error code* *Loopback location* *Supplementary capabilities* *Cause identifier*
->M_PARAMETER_LEN	Length of Paremeter, Octet 7h

6.2.14.1. Parameter Field for Parameter Identifier: #SERVICE_PROFILE_ID

Selector Name and related Constants	AT&T Reference
->M_SPID_VALUE	SPID value, Octet 7i *

6.2.14.2. Parameter Field for Parameter Identifier: #ENDPOINT_ID

Selector Name and related Constants	AT&T Reference
->M_USID	USID, Octet 7i.1
->M_FLAG #RESTRICTED_SERVICE #FULL_SERVICE	Flag, Octet 7i.2 *Restricted service* *Full service*
->M_TID	TID, Octet 7i.2

6.2.14.3. Parameter Field for Parameter Identifier: #CALL_REF_ID

Selector Name and related Constants	AT&T Reference
->M_FLAG #ORIG_SIDE #DEST_SIDE	Flag, Octet 7i *Origination side* *Destination side*
->M_CALL_REF	Call reference value, Octet 7i

ISDN LAYER 3 NATIONAL VARIANT (VN3)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	November, 1992
Engeneering	:	AUT E33 D / Zbk
Quality control	:	MWB KQP R.

6.2.14.4. Parameter Field for Parameter Identifier: #CALL_STATE_ID

Selector Name and related Constants	AT&T Reference
->M_CALL_STATE	Call State, Octet 7i
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#RELEASE_REQUEST	*Release Request*

6.2.14.5. Parameter Field for Parameter Identifier: #CHANNEL_ID

Selector Name and related Constants	AT&T Reference
->M_INT_PRESENT	Interface id. present, Octet 7i
#IMPLICIT	*Implicitly identifier*
->M_INT_TYPE	Interface Type, Octet 7i
#BASIC_INTERFACE	*Basic Interface*
->M_PREF / EXCL	Preferred / Exclusive, Octet 7i
#PREFERRED	*Preferred*
#EXCLUSIVE	*Exclusive*
->M_DCHANNEL	D-Channel Idicator, Octet 7i
#NOT_D_CHANNEL	*Not D-Channel*
#D_CHANNEL	*D-Channel*
->M_INFO_CHAN_SEL	Info Channel Selection, Octet 7i
#NO_CHANNEL	*No Channel*
#B1_CHANNEL	*B1 Channel*
#B2_CHANNEL	*B2 Channel*
#ANY_CHANNEL	*Any Channel*

6.2.14.6. Parameter Field for Parameter Identifier: #SERVICE_STATE

Selector Name and related Constants	AT&T Reference
->M_SERVICE_STATE	Service State, Octet 7i
#IN_SERVICE	*In service*
#OUT_OF_SERVICE	*Out of service*

1.2.14.7. Parameter Field for Parameter Identifier: #SERVICE_MSG_ERROR

Selector Name and related Constants	AT&T Reference
->M_ERROR_CODE #TEST_IN_PROGRESS	Error Code, Octet 7i *Test in Progress*

1.2.14.8. Parameter Field for Parameter Identifier: #LOOPBACK_LOCATION

Selector Name and related Constants	AT&T Reference
->M_LOOPBACK	Loopback Location, Octet 7i *

1.2.14.9. Parameter Field for Parameter Identifier: #SUPPL_CAPABILITIES

Selector Name and related Constants	AT&T Reference
->M_DISPLAY #SUPPORTED #NOT_SUPPORTED	Display, Octet 7i.1 *Display capabilities supported* *Display capabilities not supported*
->M_SUPPL_DATA #SUPPORTED #NOT_SUPPORTED	Suppl. Data, Octet 7i.1 *Supplementary data services supported* *Supplementary data services not supported*
->M_FC #SUPPORTED #NOT_SUPPORTED	FC, Octet 7i.1 *Flexible call offering supported* *Flexible call offering not supported*
->M_TMF #SUPPORTED #NOT_SUPPORTED	TMF, Octet 7i.1 *Terminal management feature supported* *Terminal manag. feature not supported*
->M_ASSOC #SUPPORTED #NOT_SUPPORTED	ASSOC, Octet 7i.1 *Key system protocol supported* *Key system protocol not supported*
->M_LOOP #SUPPORTED #NOT_SUPPORTED	Loop, Octet 7i.2 *Loopback capabilities supported* *Loopback capabilities not supported*
->M_SVC #SUPPORTED #NOT_SUPPORTED	Svc., Octet 7i.2 *Endpoint service msg procedures supported* *Endpoint service msg procedures not supp.*
->M_INIT #SUPPORTED #NOT_SUPPORTED	Init., Octet 7i.2 *Endpoint initialization supported* *Endpoint initialization not supported*

1.2.14.10. Parameter Field for Parameter Identifier: #CAUSE_ID

Selector Name and related Constants	AT&T Reference
->M_CAUSE	Cause Value, Octet 7i
#CHANNEL_IN_USE	*Channel in use*
#MNG_PARAM_UNKNOWN	*Provided management parameter unknown*
#INVALID_PARAM	*Invalid parameter value*

1.2.14.11. Parameter Field for Other Parameter Identifier

Selector Name and related Constants	AT&T Reference
->M_PARAMETER_VAL	Parameter Value, Octet 7i *

Note that only **one** Parameter may be coded/decoded in a Management IE.

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

3.1. CODESET 0 3-1

3.2. CODESET 6 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. CODESET 0 IEs 6-1
6.1.1 Shift IE (I#SHIFT) 6-1
6.1.2 Congestion Level IE (I#CONG_LEVEL) 6-1
6.1.3 Segmented Message IE (I#SEG_MSG) 6-1
6.1.4 Bearer Capability IE (I#BEARER_CAP) 6-2
6.1.5 Cause IE (I#CAUSE) 6-2
6.1.6 Call Identity IE (I#CALL_ID) 6-5
6.1.7 Call State IE (I#CALL_STATE) 6-5
6.1.8 Channel Identification IE (I#CHANNEL_ID) 6-6
6.1.9 Facility IE (I#FACIL) 6-7
6.1.10 Progress Indicator IE (I#PROGRESS_IND) 6-10
6.1.11 Network-Specific Facility IE (I#NETWORK_FACIL) 6-10
6.1.12 Notification Indicator IE (I#NOTIFIC_IND) 6-10
6.1.13 Display IE (I#DISPLAY) 6-11
6.1.14 Date/Time IE (I#DATE) 6-11
6.1.15 Keypad IE (I#KEYPAD) 6-11
6.1.16 Signal IE (I#SIGNAL) 6-12
6.1.17 Calling Number IE (I#CALLING_NUM) 6-13
6.1.18 Calling Subaddress IE (I#CALLING_SAD) 6-13
6.1.19 Called Number IE (I#CALLED_NUM) 6-14
6.1.20 Called Subaddress IE (I#CALLED_SAD) 6-14
6.1.21 Redirecting Number IE (I#REDIRECTING_NUM) 6-15
6.1.22 Restart Indicator IE (I#RESTART_IND) 6-15
6.1.23 Low Layer Compatability IE (I#LOW_LAY_COMP) 6-16
6.1.24 High Layer Compatability IE (I#HI_LAY_COMP) 6-20
6.1.25 User User Information IE (I#UU_INFO) 6-20

6.2. CODESET 6 IEs 6-21
6.2.1 Terminal Capability IE (I#TERM_CAPAB) 6-21
6.2.2 User User Facility IE (I#UU_FACIL) 6-21

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, section 3 lists the **Information Element** (IE) Identifiers, while section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC / *COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, section10 [access] or the ISDN EMULATION / SIMULATION Manual, section 6 [set]).

2. MESSAGE TYPE IDENTIFIER

Message Identifier	VN3 Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknowledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknowledge
M#RES	Resume
M#RES_ACK	Resume Acknowledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknowledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknowledge
M#SEGMENT	Segmented Message
M#CON_CON	Congestion Control
M#FAC	Facility
M#FAC_ACK	Facility Acknowledge
M#FAC_REJ	Facility Reject
M#REG	Register
M#REG_ACK	Register Acknowledge
M#NOTIFY	Notify
M#REG_REJ	Register Reject
M#INFO	Info
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

3. INFORMATION ELEMENT TYPE IDENTIFIER

3.1. CODESET 0

IE Identifier	VN3 Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information

3.2. CODESET 6

IE Identifier	VN3 Reference
I#TERM_CAPAB	Terminal Capabilities
I#UU_FACIL	User-User Facility

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,		
OCTET_3.1,	OCTET_3.2,	OCTET_3.3,		
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,			
OCTET_6.1,	OCTET_6.2,	OCTET_6.3,	OCTET_6.4,	
OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,	
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T302	#T302	–	15
Timer T303	#T303	10	4
Timer T304	#T304	60	–
Timer T305	#T305	4	30
Timer T307	#T307	–	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	60	10
Timer T313	#T313	4	–
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	–	120
Timer T318	#T318	10	–
Timer T319	#T319	4	–
Timer T322	#T322	4	4
Timer T399	#T399	10	–
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. CODESET 0 IEs

6.1.1. Shift IE (I#SHIFT)

Selector Name and related Constants	VN3 Reference
->SH_TYPE #NON_LOCKING #LOCKING	Locking/Non-locking Shift Type *Non Locking* *Locking*
->SH_CODESET #CODESET0 #CODESET6	Codeset Specified in Shift IE *I.451 (Q.931) IE* *Local Network Specific IE*

6.1.2. Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	VN3 Reference
->CL_CONGESTION_LEVEL #RECEIVER_READY #RECEIVER_NOT_READY	Congestion Level *Receiver Ready* *Receiver not Ready*

6.1.3. Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.1.4. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5,

Selector Name and related Constants	VN3 Reference
->BC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *3.1 kHz Audio*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s*
->BC_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->BC_L1_PROTOCOL #G.711_ALAW	Layer 1 Protocol Identifier, Octet 5 *Recommendation G.711 A-law*

6.1.5. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->C_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->C_LOCATION #USER #LOCAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local private network* *Local public network* *Transit network* *Remote public network* *International network* *Network beyond interworking point*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned number*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NUMBER_UNAWARDED	*Number not awarded*
#NON_SELECTED_CLEAR	*Non selected user clearing*
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
->C_CAUSE_CLASS	Cause Class, Octet 4
#RESOURCE_UNAVAIL	*Resource not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NO_CHANNEL_AVAIL	*No Circuit/Channel Available*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access information discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#QUALITY_UNAVAIL	*Quality of service unavailable*
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer capability not authorized*
#BEARER_UNAVAIL	*Bearer capability not available*
#CALL_RESTRICTED	*Call restricted*
#TRANSFER_REJECT	*Transfer of national call rejected*
#SERVICE_NOT_SUBSCRIBED	*Service not subscribed*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#CHANNEL_TYPE_UNIMPL	*Channel Type not Implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTENT #CALL_ID_NONEXISTENT #CALL_ID_IN_USE #NO_CALL_SUSPENDED #CALL_CLEARED #INCOMPATIBLE_DEST #INVALID_PARAM #INVALID_MESSAGE_UNSPEC	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Call Identity does not Exist* *Call Identity in Use* *No Call suspended* *Call has been cleared* *Incompatible Destination* *Invalid parameter of requested facility* *Invalid Message*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #INVALID_IE_LENGTH #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on timer expiry* *Invalid Length of mandatory IE* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #SERVICE_SUPP_UNAVAIL #TRANSFER_IMPOSSIBLE #PASSAGE_IN_POSITION #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Service support not available* *Transfer of info not possible* *Passage in position of conversation* *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.1.6. Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.1.7. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Received*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*

6.1.8. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->CID_INT_PRESENT #IMPLICIT	Interface Identifier Present, Octet 3 *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Primary rate Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_CODING_STANDARD #CCITT	Coding Standard, Octet 4 *CCITT*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 4 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS	Channel or Map Type, Octet 4 *B-Channel Units*
->CID_NUMBER	Channel number, Octet 5

6.1.9. Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	VN3 Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #USER_NOT_SUBSCRIBED	Error Value, Octet 6.5 (Note 3) *User not subscribed*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.1.10. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->PI_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->PI_LOCATION #USER #LOCAL_PRIVATE #LOCAL_PUBLIC #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local private network* *Local public network* *Remote public network* *Remote private network* *International network* *Network beyond interworking point*
->PI_DESCRIPTION #NOT_END_TO_END #DEST_NON_ISDN #ORIG_NON_ISDN #RETURNED_TO_ISDN #INBAND_INFO_AVAIL	Progress Description, Octet 4 *Call is not end-to-end ISDN* *Destination Address is non-ISDN* *Origination Address is non-ISDN* *Call has returned to the ISDN* *In-band info now available*

6.1.11. Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.1.12. Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change*

6.1.13. Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->D_DISPLAY	Display, Octet 3 *

6.1.14. Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8

Selector Name and related Constants	VN3 Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7
->DT_SECOND	Second, Octet 8

6.1.15. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.16. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*

6.1.17. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #SUBSCRIBER	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Subscriber Number*
->CLGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED #NUMBER_UNAVAIL	Presentation Indicator, Octet 3 *Presentation allowed* *Presentation restricted* *Number not available due to interworking*
->CLGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.1.18. Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.1.19. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDN_NUMBER_TYPE #BLOCK_NUMBER #OVERLAP_SEND_NUMBER	Type of Number, Octet 3 *Block numbering* *Overlap sending*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.20. Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.1.21. Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	VN3 Reference
->RDGN_NUMBER_TYPE #INTERNATIONAL #NATIONAL	Type of Number, Octet 3 *Other International Standards* *National Standard*
->RDGN_NUMBERING_PLAN #ISDN_PLAN	Numbering Plan Identification, Octet 3 *ISDN Numbering Plan E.164*
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #NETWORK_PROVIDED	Screening indicator, Octet 3a *Network-Provided*
->RDGN_REASON	Reason for redirection, Octet 3b
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.1.22. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->RI_CLASS #INDICATED_CHANNEL #SINGLE_INTERFACE #ALL_INTERFACES	Class of Channel or Interface, Octet 3 *Indicated Channel* *Single Interface* *All Interface*

6.1.23. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	VN3 Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #64KBIT/S #2x64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s* *2 x 64 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC	Symmetry, Octet 4b *Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B #64KBIT/S #2x64KBIT/S	Information Transfer Rate (D->O), Octet 4b *64 kbit/s* *2 x 64 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*

Selector Name and related Constants	VN3 Reference
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#38.4KBIT/S	*38.4kbit/s (not ETSI)*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#38.4KBIT/S_V14	*38.4kbit/s (V14, neither ETSI nor CCITT)*
#57.6KBIT/S_V14	*57.6kbit/s (V14, neither ETSI nor CCITT)*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*

Selector Name and related Constants	VN3 Reference
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*Not Used*
#INT_8KBIT/S	*8kbit/s*
#INT_16KBIT/S	*16kbit/s*
#INT_32KBIT/S	*32kbit/s*
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*Flow Control not required*
#REQUIRED	*Flow Control Required*
->LL_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot Accept Flow Control*
#ACCEPT	*Can Accept Flow Control*
->LL_V120_RATE_ADAPT	V.120 rate adaption field, Octet 5b.2 (Note 2)
->LL_STOP_BITS	Number of Stop Bits, Octet 5c
#NOT_USED	*Not used*
#1_STOP_BIT	*1 Bit*
#1.5_STOP_BITS	*1.5 Bits*
#2_STOP_BITS	*2 Bits*
->LL_DATA_BITS	Number of Data Bits, Octet 5c
#NOT_USED	*Not used*
#5_DATA_BITS	*5 Bits*
#7_DATA_BITS	*7 Bits*
#8_DATA_BITS	*8 Bits*
->LL_PARITY	Parity Information, Octet 5c
#ODD_PARITY	*Odd*
#EVEN_PARITY	*Even*
#NO_PARITY	*None*
#FORCED_TO_0	*Forced to 0*
#FORCED_TO_1	*Forced to 1*
->LL_DUPLEX_MODE	Duplex mode, Octet 5d
#HALF_DUPLEX	*Half duplex*
#FULL_DUPLEX	*Full duplex*

Selector Name and related Constants	VN3 Reference
->LL_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
->LL_LAYER2_ID	Layer 2 Identification, Octet 6
#LAYER2	*User Info Layer 2 Protocol*
->LL_L2_PROTOCOL	Layer 2 Protocol Identifier, Octet 6
#ISO_1745	*Basic mode ISO 1745*
#Q.921	*Recommendation Q.921 (i.441)*
#X.25_LINK	*Recommendation X.25, Link level*
#X.25_MULTILINK	*Recommendation X.25 Multilink*
#EXTENDED_LAPB	*Extended LAPB; half duplex (T.71)*
#HDLC_ARM	*HDLC ARM (ISO 4335)*
#HDLC_NRM	*HDLC NRM (ISO 4335)*
#HDLC_ABM	*HDLC ABM (ISO 4335)*
#LAN_LLC	*LAN Logical link control (ISO 8802/2)*
#X.75	*Recommendation X.75, single link procedure*
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID	Layer 3 identification, Octet 7
#LAYER3	*User Info Layer 3 Protocol*
->LL_L3_PROTOCOL	Layer 3 Protocol Identifier, Octet7
#Q.931	*Recommendation Q.931 (I.451)*
#X.25_PACKET	*Recommendation X.25, Packet Layer*
#ISO_8208	*ISO 8208*
#ISO_8348	*ISO 8348*
#ISO_8473	*ISO 8473*
#T.70	*Recommendation T.70, min. network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30 or #NON_CCITT

Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.1.24. High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->HL_CODING_STANDARD #CCITT #NATIONAL	Coding Standard, Octet 3 *CCITT* *National Standard*
->HL_INTERPRETATION #NATIONAL_INTERPRET #CCITT/CEPT	Interpretation, Octet 3 *National* *CCITT / CEPT*
->HL_PRESENTATION #PROTOCOL_PROFILE	Presentation Method of Profile, Octet 3 *High Layer Protocol Profile*
->HL_CHARACTER #TELEPHONY #FAX_GROUP_2/3 #FAX_GROUP_4 #MIXED_MODE #PROCESSABLE #TELETEX #VIDEOTEX #VIDEOTEX_INTERNAT #TELEX #MHS #OSI_APPLICATION #AUDIOVISUAL #UNKNOWN #AUDIOGRAPHY #AUDIOVIDEOTEX #NOT_NORMAL	High Layer Characteristics Id., Octet 4 *Telephony Rec. G.711* *Fax Group 2/3 Rec. T.62* *Fax Group 4 Rec. T503* *Mixed Mode Rec. T.501* *Teletex (base mode) Rec. F.220* *Teletex Rec. T.62, T.70* *Videotex Rec. T.504* *International Videotex F.300, T.101* *Telex* *Message Handling Systems (MHS)* *OSI Application Rec. X.200* *Audiovisual Rec. F.720, H.242* *Unknown* *Audiography* *Audiovideotex* *Applications not normalized*

6.1.25. User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UU_PROTOCOL_DISCR #IA5_CHAR #NT_NT_FACILITY #TRANSGROUP	Protocol discriminator, Octet 3 *IA5 characters* *Network to Network facility* *Transgroup facilities*
->UU_USER_INFO	User-User Information, Octet 4 *

6.2. CODESET 6 IEs

6.2.1. Terminal Capability IE (I#TERM_CAPAB)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->TC_CODING_STANDARD #NATIONAL	Coding Standard, Octet 3 *National Standard*
->TC_DESCRIPTION #NAT_TERMINAL	Capability Description, Octet 3 *National terminal mode*

6.2.2. User User Facility IE (I#UU_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UF_PROTOCOL_DISCR #NT_NT_FACILITY #MINI_MSG #TRANSGROUP	Coding Standard, Octet 3 *Network to Network facility* *Mini message* *Transgroup facilities*
->UF_USER_FACIL	User facility, Octet 4 *

ISDN LAYER 3
NATIONAL VARIANT
(Bellcore SR-NWT-001953)

User Manual for
Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2 Rev.1
Revision Date	:	November, 1993
Engineering	:	AUT E33 D / SK
Quality control	:	MWB KQP R.

1	**INTRODUCTION**	1–1
2	**MESSAGE TYPE IDENTIFIER**	2–1
2.1	ONE–OCTET MESSAGES	2–1
2.2	TWO–OCTET MESSAGES	2–1
3	**INFORMATION ELEMENT TYPE IDENTIFIER**	3–1
3.1	CODESET 0 IE's	3–1
3.2	CODESET 5 IE's	3–2
3.3	CODESET 6 IE's	3–2
4	**OCTET IDENTIFIER**	4–1
5	**LAYER 3 TIMER**	5–1
6	**ACCESSING INFORMATION ELEMENT FIELD VALUES**	6–1
6.1	CODESET 0 IE's	6–1
6.1.1	LOCKING SHIFT IE (I#SHIFT)	6–1
6.1.2	BEARER CAPABILITY IE (I#BEARER_CAP)	6–1
6.1.3	CAUSE IE (I#CAUSE)	6–3
6.1.4	CALL STATE IE (I#CALL_STATE)	6–5
6.1.5	CHANNEL IDENTIFICATION IE (I#CHANNEL_ID)	6–7
6.1.6	PROGRESS INDICATOR IE (I#PROGRESS_IND)	6–8
6.1.7	NOTIFICATION INDICATOR IE (I#NOTIFIC_IND)	6–9
6.1.8	KEYPAD IE (I#KEYPAD)	6–9
6.1.9	INFORMATION REQUEST IE (I#INFO_REQ)	6–9
6.1.10	SIGNAL IE (I#SIGNAL)	6–10
6.1.11	FEATURE ACTIVATION IE (I#FEATURE_ACT)	6–10
6.1.12	FEATURE INDICATION IE (I#FEATURE_IND)	6–11
6.1.13	SERVICE PROFILE IDENTIFICATION (I#SERV_PROF_ID)	6–11
6.1.14	ENDPOINT IDENTIFIER (I#END_ID)	6–11
6.1.15	INFORMATION RATE IE (I#INFO_RATE)	6–12
6.1.16	END–TO–END TRANSIT DELAY IE (I#END_DELAY)	6–13
6.1.17	TRANSIT DELAY SELECTION AND INDICATION IE (I#TRANS_DEL)	6–14
6.1.18	PACKET LAYER BINARY PARAMETERS IE (I#PACKET_BIN)	6–14
6.1.19	PACKET LAYER WINDOW SIZE IE (I#PACKET_WIN)	6–14
6.1.20	PACKET SIZE IE (I#PACKET_SIZE)	6–15
6.1.21	CLOSED USER GROUP (I#CLOSED_U_GROUP)	6–15
6.1.22	REVERSE CHARGING INDICATION (I#REV_CHARG_IND)	6–15
6.1.23	CALLING PARTY NUMBER IE (I#CALLING_NUM)	6–16
6.1.24	CALLING PARTY SUBADDRESS IE (I#CALLING_SAD)	6–16
6.1.25	CALLED PARTY NUMBER IE (I#CALLED_NUM)	6–17
6.1.26	CALLED PARTY SUBADDRESS IE (I#CALLED_SAD)	6–17
6.1.27	REDIRECTING NUMBER IE (I#REDIRECTING_NUM)	6–18
6.1.28	REDIRECTION NUMBER IE (I#REDIRECTION_NUM)	6–18
6.1.29	TRANSIT NETWORK SELECTION IE (I#TRANS_NW_SEL)	6–19
6.1.30	LOW LAYER COMPATIBILITY IE (I#LOW_LAY_COMP)	6–20
6.1.31	HIGH LAYER COMPATIBILITY IE (I#HI_LAY_COMP)	6–25
6.1.32	USER USER INFORMATION IE (I#UU_INFO)	6–26
6.2	CODESET 5 IE's	6–26
6.2.1	DISPLAY TEXT IE (I#DISPLAY_TEXT)	6–26
6.2.2	OPERATOR SYSTEM ACCESS IE (I#OPERATOR)	6–26
6.2.3	SOFT KEY CONTROL IE (I#SOFTKEY)	6–27
6.3	CODESET 6 IE's	6–27
6.3.1	CALL APPEARANCE IE (I#CALL_APP)	6–27
6.3.2	REDIRECTING SUBADDRESS IE (I#REDIRECTING_SAD)	6–27
6.3.3	REDIRECTION NUMBER IE (I#REDIRECTION_NUM6)	6–28

1 INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

This National ISDN Variant is based on specification:

Bellcore SR-NWT-001953
Issue 1, June 1991
Revision 1, December 1991
Generic Guidelines for ISDN Terminal Equipment on Basic Access Interfaces

2 MESSAGE TYPE IDENTIFIER

2.1 ONE-OCTET MESSAGES

Message Identifier	Bellcore Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknowledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknowledge
M#HOLD	Hold
M#HOLD_ACK	Hold Acknowledge
M#HOLD_REJ	Hold Reject
M#RET	Retrieve
M#RET_ACK	Retrieve Acknowledge
M#RET_REJ	Retrieve Reject
M#DISC	Disconnect
M#REL	Release
M#REL_COM	Release Complete
M#INFO	Information
M#NOTIFY	Notify
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

2.2 TWO-OCTET MESSAGES

Message Identifier	Bellcore Reference
M#KEY_HOLD	Key Hold
M#KEY_REL	Key Release
M#KEY_SETUP	Key Setup
M#KEY_SETUP_ACK	Key Setup Acknowledge

3 INFORMATION ELEMENT TYPE IDENTIFIER

3.1 CODESET 0 IE's

IE Identifier	Bellcore Reference
I#SHIFT	Locking Shift
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#PROGRESS_IND	Progress Indicator
I#NOTIFIC_IND	Notification Indicator
I#KEYPAD	Keypad
I#INFO_REQ	Information Request
I#SIGNAL	Signal
I#FEATURE_ACT	Feature Activation
I#FEATURE_IND	Feature Indication
I#SERV_PROF_ID	Service Profile Identification
I#END_ID	Endpoint Identifier
I#INFO_RATE	Information Rate
I#END_DELAY	End–to–End Transit Delay
I#TRANS_DEL	Transit Delay Selection and Indication
I#PACKET_BIN	Packet Layer Binary Parameter
I#PACKET_WIN	Packet Binary Window Size
I#PACKET_SIZE	Packet Size
I#CLOSED_U_GROUP	Closed User Group
I#REV_CHARG_IND	Reverse Charging Indication
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#REDIRECTION_NUM	Redirection Number
I#TRANS_NW_SEL	Transit Network Selection
I#LOW_LAY_COMP	Low Layer Compatibility
I#HI_LAY_COMP	High Layer Compatibility
I#UU_INFO	User–User Information

3.2 CODESET 5 IE's

IE Identifier	Bellcore Reference
I#DISPLAY_TEXT	Display Text
I#OPERATOR	Operator System Access
I#SOFTKEY	Soft Key Control

3.3 CODESET 6 IE's

IE Identifier	Bellcore Reference
I#CALL_APP	Call Appearance
I#REDIRECTING_SAD	Redirecting Subaddress
I#REDIRECTION_NUM6	Redirection Number

4 OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	**OCTET_3A,**	**OCTET_3B,**	**OCTET_3.1,**	**OCTET_3.2,**
OCTET_3.3,				
OCTET_4,	**OCTET_4A,**	**OCTET_4B,**		
OCTET_5,	**OCTET_5A,**	**OCTET_5B,**	**OCTET_5B.1,**	**OCTET_5B.2,**
OCTET_5C,	**OCTET_5D,**			
OCTET_6,	**OCTET_6A,**	**OCTET_6.1,**	**OCTET_6.2,**	**OCTET_6.3,**
OCTET_6.4,	**OCTET_6.5,**	**OCTET_6.6,**	**OCTET_6.7,**	**OCTET_6.8,**
OCTET_6.8.1,	**OCTET_6.8.2,**	**OCTET_6.8.3,**	**OCTET_6.8.4,**	**OCTET_6.8.5,**
OCTET_7,	**OCTET_7A,**			
OCTET_8				

5 LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T301	#T301	180	180
Timer T302	#T302	15	15
Timer T303	#T303	4	4
Timer T304	#T304	15	20
Timer T305	#T305	30	30
Timer T306	#T306	30	30
Timer T307	#T307	180	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	10	10
Timer T312	#T312	6	6
Timer T313	#T313	4	4
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	60	60
Timer T318	#T318	4	4
Timer T319	#T319	4	4
Timer T320	#T320	30	30
Timer T321	#T321	30	30
Timer T322	#T322	4	4
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6 ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1 CODESET 0 IE's

6.1.1 LOCKING SHIFT IE (I#SHIFT)

Selector Name and related Constants	Bellcore Reference
–>SH_TYPE	Shift–type
#LOCKING	*Locking*
–>SH_CODESET	Codeset Specified in Shift IE
#CODESET5	*National Use IE*
#CODESET6	*Local Network Specific IE*

6.1.2 BEARER CAPABILITY IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4,OCTET_5, OCTET_5A, OCTET_6, OCTET_7

Selector Name and related Constants	Bellcore Reference
–>BC_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*Q.931 Standard*
–>BC_TRANSFER_CAP	Information Transfer Capability, Octet 3
#SPEECH	*Speech*
#UNRESTRICTED	*Unrestricted Digital Information*
#3.1KHZ_AUDIO	*3.1 kHz Audio*

Selector Name and related Constants	Bellcore Reference
–>BC_TRANSFER_MODE_RATE **#BC_64KBIT/S** **#BC_PACKET_MODE**	Transfer Mode and Rate, Octet 4 *64 kbit/s, Circuit Mode* *Packet Mode*
–>BC_L1_PROTOCOL **#RATE_ADAPTION** **#BC_G.711_ULAW**	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption* *Recommendation G.711 u–law*
–>BC_USER_RATE **#56KBIT/S**	User Rate, Octet 5A *56 kbit/s (Rec. V.110 / I.463)*
–>BC_L2_PROTOCOL **#BC_Q.921** **#BC_X.25_LINK**	Layer 2 Protocol Identifier, Octet 6 *LAPD (Q.921 / I.441)* *LAPB (X.25, link level)*
–>BC_L3_PROTOCOL **#BC_X.25_PACKET**	Layer 3 Protocol Identifier, Octet 7 *X.25, packet layer*

6.1.3 CAUSE IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	Bellcore Reference
->C_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Network specific*
->C_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVATE	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#TRANSIT	*Transit Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL0	*Normal Event Class 0*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT or #NATIONAL	
#UNASSIGNED_NUMBER	*Unassigned Number*
#NO_ROUTE_TO_TRANSIT	*No Route to Transit Network*
#NO_ROUTE_TO_DEST	*No Route to Destination*
#CHANNEL_UNACCEPTABLE	*Channel Unacceptable*
#CALL_AWARDED	*Call awarded*
#VACANT_CODE	*Vacant code*
#PREF0_DIALED	*Prefix 0 dialed in error*
#PREF1_DIALED	*Prefix 1 dialed in error*
#PREF1_NOT_DIALED	*Prefix 1 not dialed*
#EXCESSIVE_DIGITS	*Excessive digits received*
for Coding Standard #NETWORK_SPECIFIC	
#CALL_PROC	*Call proceeding*
#SERVICE_DENIED	*Service denied*

Selector Name and related Constants	Bellcore Reference
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NON_SELECTED_CLEAR	*Non selected user clearing*
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility Rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
for Coding Standard #NETWORK_SPECIFIC	
#SPEC_INTERC_ANN	*Special intercept announcement*
#SPEC_INTERC_ANN_UNDEF	*Special intercept announcement*
#SPEC_INTERC_ANN_UNASS	*Special intercept announcement*
#SPEC_INTERC_ANN_BLOCK	*Special intercept announcement*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NETWORK_CONGEST	*Network Congestion*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#CIRCUIT_CONG	*Circuit/Channel congestion*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access Information Discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer Capability not Authorized*
#BEARER_UNAVAIL	*Bearer Capability not Available*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
for Coding Standard #NETWORK_SPECIFIC	
#INCOMP_CALL_TYPE	*Call type incompat. with service*
#OPERATION_VIOLATED	*Service Operation violated*

Selector Name and related Constants	Bellcore Reference
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INVALID_MESSAGE	*Invalid Message*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#INVALID_CALL_REF	*Invalid Call Reference Value*
#INCOMPATIBLE_DEST	*Incompatible Destination*
->C_CAUSE_CLASS	Cause Class, Octet 4
#PROTOCOL_ERROR	*Protocol Error*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#MAND_IE_MISSING	*Mandatory IE is Missing*
#MESSAGE_TYPE_UNIMPL	*Message Type Non−existing*
#IE_UNIMPL	*IE Non−existing*
#INVALID_IE_CONTENTS	*Invalid IE Contents*
#MESSAGE_UNDEFINED	*Message not compatible with state*
#TIMER_EXPIRY	*Recovery on Timer Expiry*
#PROTOCOL_ERROR_UNSPEC	*Protocol error, unspecified*
for Coding Standard #NETWORK_SPECIFIC	
#THRESHOLD_EXCEEDED	*Protocol error threshold exceeded*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INTERWORKING	*Interworking*
->C_CAUSE_VALUE	Cause Value, Octet 4
for Coding Standard #CCITT	
or #NATIONAL	
#INTERWORK_UNSPEC	*Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.1.4 CALL STATE IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->CS_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*Q.931 Standard*
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnect Request*
#DISC_INDICATION	*Disconnect Indication*
#RELEASE_REQUEST	*Release Request*

6.1.5 CHANNEL IDENTIFICATION IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
–>CID_INT_PRESENT	Interface Identifier Present, Octet 3
#IMPLICIT	*Implicitly Identified*
–>CID_INT_TYPE	Interface Type, Octet 3
#BASIC_INTERFACE	*Basic Interface*
–>CID_PREF/EXCL	Preferred/Exclusive, Octet 3
#EXCLUSIVE	*Exclusive*
#PREFERRED	*Preferred*
–>CID_DCHANNEL	D–channel Indicator, Octet 3
#D_CHANNEL	*D–Channel Indentified*
#NOT_D_CHANNEL	*Not D–Channel*
–>CID_INFO_CHAN_SEL	Information Channel Selection, Octet 3
#NO_CHANNEL	*No Channel*
#B1_CHANNEL	*B1 Channel*
#B2_CHANNEL	*B2 Channel*
#ANY_CHANNEL	*Any Channel*

6.1.6 PROGRESS INDICATOR IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->PI_CODING_STANDARD	Coding Standard, Octet 3.2
#CCITT	*CCITT*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->PI_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVATE	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#TRANSIT	*Transit network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->PI_DESCRIPTION	Progress Description, Octet 4
#NOT_END_TO_END	*Call is not end-to-end ISDN*
#DEST_NON_ISDN	*Destination Address is non-ISDN*
#ORIG_NON_ISDN	*Origination Address is non-ISDN*
#INBAND_INFO_AVAIL	*In-band Info now Available*
#DELAY_IN_RESPONSE	*Delay in response at destination*

6.1.7 NOTIFICATION INDICATOR IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->NI_DESCRIPTION	Notification Descriptor, Octet 3
#USER_BRIDGED	*User bridged onto call*
#ACB_MON_DISCONT	*ACB monitoring discontinued*
#CALL_ON_HOLD	*Call on hold*
#MON_USER_IDLE	*monitored user idle*
#REMOTE_HOLD	*remote hold*
#REMOTE_HOLD_REL	*remote hold released*
#CALL_FORWARDED	*Call is forwarded*
#PRIVACY_ENABLED	*Privacy enabled*
#PRIVACY_DISABLED	*Privacy disabled*
#RETRIEVED_FROM_HOLD	*Call retrieved from hold*

6.1.8 KEYPAD IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.9 INFORMATION REQUEST IE (I#INFO_REQ)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->IREQ_IND	Information Request Indicator
#REQUEST_COMPLETED	*Information request completed*
#ADD_INFO	*Prompt for additional information*
->IREQ_INFO_TYPE	Type of information
#NOT_DEFINED	*Undefined*
#AUTHORIZ_CODE	*Authorization code*
#ADDRESS_DIGITS	*Address digits*
#TERMINAL_ID	*Terminal identification*

6.1.10 SIGNAL IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_OFF	*Alerting off*
#RECALL_ON	*Recall dial tone on*
#BARGE_IN	*Barge–in tone*
#INCOMING_ADDITION	*Incoming additional call*
#PRIORITY_ADDITION	*Priority additional call*
#EXPENSIVE_ROUTE	*Expensive route warning*

6.1.11 FEATURE ACTIVATION IE (I#FEATURE_ACT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A

Selector Name and related Constants	Bellcore Reference
->FA_ID_NUMBER	Feature identifier number, Octet 3
->FA_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a

6.1.12 FEATURE INDICATION IE (I#FEATURE_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	Bellcore Reference
->FI_ID_NUMBER	Feature identifier number, Octet 3
->FI_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a
->FI_STATUS_IND **#DEACTIVATED** **#ACTIVATED** **#PROMPT** **#PENDING**	Status indicator, Octet 4 *Idle* *Active* *Prompt* *Pending*

6.1.13 SERVICE PROFILE IDENTIFICATION (I#SERV_PROF_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->SPI_SPID	Service Profile Identifier, Octet 3 *

6.1.14 ENDPOINT IDENTIFIER (I#END_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->EI_USER_ID	User Service Identifier, Octet 3
->EI_INTERPRETER **#TID_AND_USID** **#TID_NOT_USID**	Interpreter, Octet 4 *TID and USID* *USID*
->EI_TID	Terminal Identifier, Octet 4

6.1.15 INFORMATION RATE IE (I#INFO_RATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6

Selector Name and related Constants	Bellcore Reference
–>IR_INCOMING	Incoming information rate, Octet 3
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
–>IR_OUTGOING	Outgoing information rate, Octet 4
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*

Selector Name and related Constants	Bellcore Reference
–>IR_MIN_INCOMING	Minimum incoming info rate, Octet 5
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
–>IR_MIN_OUTGOING	Minimum outgoing info rate, Octet 6
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*

6.1.16 END–TO–END TRANSIT DELAY IE (I#END_DELAY)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B

Selector Name and related Constants	Bellcore Reference
–>ED_CUM_DELAY_3	Cumultative trans. del. val., bit 16–15 (Octet 3)
–>ED_CUM_DELAY_3A	Cumultative trans. del. val., bit 14–8 (Octet 3a)
–>ED_CUM_DELAY	Cumultative transit delay value, Octet 3b
–>ED_REQ_DELAY_4	Requested trans. del. val., bit 16–15 (Octet 4)
–>ED_REQ_DELAY_4A	Requested trans. del. val., bit 14–8 (Octet 4a)
–>ED_REQ_DELAY	Requested transit delay value, Octet 4b
–>ED_MAX_DELAY_5	Maximum trans. del. val., bit 16–15 (Octet 5)
–>ED_MAX_DELAY_5A	Maximum trans. del. val., bit 14–8 (Octet 5a)
–>ED_MAX_DELAY	Maximum transit delay value, Octet 5b

6.1.17 TRANSIT DELAY SELECTION AND INDICATION IE (I#TRANS_DEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B

Selector Name and related Constants	Bellcore Reference
->TD_SEL/IND_3	Transit delay sel/ind val., bit 15–14 (Octet 3)
->TD_SEL/IND_3A	Transit delay sel/ind val., bit 13–7 (Octet 3a)
->TD_SEL/IND_3B	Transit delay sel/ind val., bit 0–6 (Octet 3b)

6.1.18 PACKET LAYER BINARY PARAMETERS IE (I#PACKET_BIN)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->PB_FAST_SELECT	Fast select, Octet 3
#NOT_REQUESTED_0	*Not requested*
#NOT_REQUESTED_1	*Not requested*
#REQUESTED_NOT_REST	*Requested, no restriction*
#REQUESTED_REST	*Requested, restriction*
->PB_EXP_DATA	Expedited data, Octet 3
#NO_REQUEST	*No request*
#REQUEST_IND	*Request indicated*
->PB_DELIVERY_CONF	Delivery confirmation, Octet 3
#LINK_BY_LINK	*Link–by–link confirmation*
#END_TO_END	*End–to–end confirmation*
->PB_MODULUS	Modulus, Octet 3
#MODULUS_8	*Modulus 8 sequencing*
#MODULUS_128	*Modulus 128 sequencing*

6.1.19 PACKET LAYER WINDOW SIZE IE (I#PACKET_WIN)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->PWS_FORWARD	Forward value, Octet 3
->PWS_BACKWARD	Backward value, Octet 4

6.1.20 PACKET SIZE IE (I#PACKET_SIZE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
–>PS_FORWARD	Forward value, Octet 3
–>PS_BACKWARD	Backward value, Octet 4

6.1.21 CLOSED USER GROUP (I#CLOSED_U_GROUP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
–>CUP_CUG_IND **#CUG_SELECTION** **#OUTGOING_ACCESS**	CUG Indication, Octet 3 *Closed User Group selection* *Outgoing access selection*
–>CUP_CUG_IND_CODE	CUG Index Code, Octet 4 *

6.1.22 REVERSE CHARGING INDICATION (I#REV_CHARG_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
–>RCI_IND **#REVERSE_REQUESTED**	Reverse Charging Indication, Octet 3 *Reverse charging requested*

6.1.23 CALLING PARTY NUMBER IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	Bellcore Reference
->CLGN_NUMBER_PLAN_TYPE **#UNKNOWN** **#ISDN_PLAN** **#DATA_PLAN** **#NATIONAL_PLAN** **#LOCAL_NUMBER** **#PRIVATE_PLAN** **#ABBREVIATED**	Type of Number/numbering Plan, Octet 3 *Unknown* *International number (E.164)* *Data Numbering Plan X.121* *National number (E.164)* *Local number (E.164)* *Private Numbering Plan* *Abbreviated number*
->CLGN_PRESENTATION **#ALLOWED_UNSCREENED** **#ALLOWED_PASSED** **#ALLOWED_FAILED** **#ALLOWED_NETWORK** **#PROHIB_UNSCREENED** **#PROHIB_PASSED** **#PROHIB_FAILED** **#PROHIB_NETWORK** **#NUMBER_UNAVAIL**	Presentation Indicator, Octet 3 *Presentation allowed, not screened* *Presentation allowed, number passed* *Presentation allowed, number failed* *Presentation Allowed, network provided* *Presentation prohib., not screened* *Presentation prohib., number passed* *Presentation prohib., number failed* *Presentation prohib., network number* *Not available due to Interworking*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.1.24 CALLING PARTY SUBADDRESS IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->CLGS_ADDRESS_TYPE **#USER_SPECIFIED** **#NSAP**	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN **#EVEN_NUMBER** **#ODD_NUMBER**	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.1.25 CALLED PARTY NUMBER IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->CLDN_NUMBER_PLAN_TYPE **#UNKNOWN** **#ISDN_PLAN** **#NATIONAL_PLAN** **#NETWORK_NUMBER** **#LOCAL_NUMBER** **#ABBREVIATED**	Type of Number and numbering Plan, Octet 3 *Unknown* *International number (E.164)* *National number (E.164)* *Network specific number* *Local number (E.164)* *Abbreviated number*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.26 CALLED PARTY SUBADDRESS IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->CLDS_ADDRESS_TYPE **#USER_SPECIFIED** **#NSAP**	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN **#EVEN_NUMBER** **#ODD_NUMBER**	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.1.27 REDIRECTING NUMBER IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	Bellcore Reference
->RDGN_NUMBER_PLAN_TYPE	Type of Number and numbering Plan, Octet 3
#UNKNOWN	*Unknown*
#ISDN_PLAN	*International number (E.164)*
#DATA_PLAN	*Data numbering plan (X.121)*
#NATIONAL_PLAN	*National number (E.164)*
#ABBREVIATED	*Abbreviated Number*
->RDGN_PRESENTATION	Presentation Indicator, Octet 3 a
#ALLOWED_NETWORK	*Presentation allowed, network-provided*
#PROHIB_NETWORK	*Presentation prohibited, network number*
#NUMBER_UNAVAIL	*Number not available*
->RDGN_REASON	Reason for redirection, Octet 3b
#UNKNOWN	*Unknown*
#CALL_FORW_BUSY	*Call forward or called DTE busy*
#CALL_FORW_NO_REPLY	*Call forward no reply*
#CALLED_OUT_OF_ORDER	*Called DTE out of order .*
#FORW_BY_CALLED	*Forwarding by called DTE*
#UNCOND_OR_SYS	*Unconditional or systematic red.*
->RDGN_NUMBER	Address, Octet 4 *

6.1.28 REDIRECTION NUMBER IE (I#REDIRECTION_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->RDIN_NUMBER_PLAN_TYPE	Type of Number and numbering Plan, Octet 3
#NATIONAL_PLAN	*National number (E.164)*
#NETWORK_NUMBER	*Network specific number*
#LOCAL_NUMBER	*Local number (E.164)*
#ABBREVIATED	*Abbreviated Number*
->RDIN_NUMBER	Address Digits, Octet 4 *

6.1.29 TRANSIT NETWORK SELECTION IE (I#TRANS_NW_SEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->TR_NET_ID_TYPE **#USER_SPECIFIED_NET** **#NATIONAL_NET**	Type of Network Identification, Octet 3 *User specified* *National network id.*
->TR_NET_ID_PLAN **#UNKNOWN** **#CARRIER_CODE**	Network Identification Plan, Octet 3 *Unknown / User specified* *Carrier id. code*
->TR_NET_ID	Network Identification, Octet 4 *

6.1.30 LOW LAYER COMPATIBILITY IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A,OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	Bellcore Reference
–>LL_CODING_STANDARD **#CCITT** **#INTERNATIONAL** **#NATIONAL** **#NETWORK_SPECIFIC**	Coding Standard, Octet 2.1 or 3 *Q.931 Standard* *Other International Standards* *National Standard* *Network specific*
–>LL_TRANSFER_CAP **#SPEECH** **#UNRESTRICTED** **#RESTRICTED** **#3.1KHZ_AUDIO** **#7KHZ_AUDIO** **#VIDEO**	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
–>LL_NEGOTIATION_IND **#NEG_NOT_POSSIBLE** **#NEG_POSSIBLE**	Negotiation indicator, Octet 3a *Out–band negotiation not possible* *Out–band negotiation possible*
–>LL_TRANSFER_MODE **#CIRCUIT_MODE** **#PACKET_MODE**	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
–>LL_TRANSFER_RATE **#CHANNEL_SIZE** **#64KBIT/S** **#2*64KBIT/S** **#384KBIT/S** **#1536KBIT/S** **#1920KBIT/S**	Information Transfer Rate, Octet 4 *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
–>LL_STRUCTURE **#DEFAULT** **#8KHZ_INTEGRITY** **#SDU_INTEGRITY** **#UNSTRUCTURED**	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
–>LL_CONFIGURATION **#POINT_TO_POINT**	Configuration, Octet 4a *Point–to–point*

Selector Name and related Constants	Bellcore Reference
–>LL_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
–>LL_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
–>LL_TRANSFER_RATE_4B	Information Transfer Rate (D–>O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2*64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
–>LL_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
–>LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110 / X.30*
#G.711_ULAW	*Recommendation G.711 u–law*
#G.711_ALAW	*Recommendation G.711 A–law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.722 / G.725 7 kHz audio*
#G.7XX_VIDEO	*Recommendation H.261 384 kbit/s video*
#NON_CCITT	*Non–CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
–>LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
–>LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In–band negotiation not possible*
#NEG_POSSIBLE	*In–band negotiation possible*
–>LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E–bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*

Selector Name and related Constants	Bellcore Reference	
#7.2KBIT/S	*7.2kbit/s*	*Recommendation V.6*
#8KBIT/S	*8kbit/s*	*Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s*	*Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s*	*Recommendation V.6*
#16KBIT/S	*16kbit/s*	*Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s*	*Recommendation V.6*
#32KBIT/S	*32kbit/s*	*Recommendation I.460*
#48KBIT/S	*48kbit/s*	*Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s*	*Recommendation V.6*
#64KBIT/S	*64kbit/s*	*Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s*	*Recommendation X.1*
#0.100KBIT/S	*0.100 kbit/s*	*Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s*	*Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0.075kbit/s*	*Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s*	*Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075 kbit/s*	*Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s*	*Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s*	*Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s*	*Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s*	*Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s*	*Recommendation V.6*
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)	
#NOT_USED	*Not used*	
#INT_8KBIT/S	*8 kbit/s*	
#INT_16KBIT/S	*16 kbit/s*	
#INT_32KBIT/S	*32 kbit/s*	
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1	
#NOT_REQUIRED	*Data with NIC not required*	
#REQUIRED	*Data with NIC required*	
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1	
#NOT_ACCEPT	*Cannot accept data with NIC*	
#ACCEPT	*Can accept data with NIC*	
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1	
#NOT_REQUIRED	*No Flow Control required*	
#REQUIRED	*Flow Control Required*	

Selector Name and related Constants	Bellcore Reference
–>LL_FLOW_CTRL_RX #NOT_ACCEPT #ACCEPT	Flow Control on RX, Octet 5b.1 *Cannot Accept Flow Control* *Can Accept Flow Control*
–>LL_RATE_HEADER #NOT_INCLUDED #INCLUDED	Rate Adaption Header, Octet 5b.2 (Note 2) *Header not included* *Header included*
–>LL_MULTI_FRAME #NOT_SUPPORTED #SUPPORTED	Multi Frame establishment, Octet 5b.2 *Not supported, only UI allowed* *Supported*
–>LL_MODE #BIT_TRANSPARENT #PROTOCOL_SENSITIVE	Mode of operation, Octet 5b.2 *Bit transparent mode* *Protocol sensitive mode*
–>LL_LLI_NEG #DEFAULT #FULL	Logical Link negotiation, Octet 5b.2 *Default, LLI=256 only* *Full protocol negotiation*
–>LL_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assignee'* *Originator is 'Assignor only'*
–>LL_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out–band negotiation, Octet 5b.2 *Negotiation is done 'Out–band'* *Negotiation is done 'In–band'*
–>LL_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of Stop Bits, Octet 5c *Not used* *1 Bit* *1.5 Bits* *2 Bits*
–>LL_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of Data Bits, Octet 5c *Not used* *5 Bits* *7 Bits* *8 Bits*
–>LL_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity Information, Octet 5c *Odd* *Even* *None* *Forced to 0* *Forced to 1*

Selector Name and related Constants	Bellcore Reference
–>LL_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
–>LL_MODEM_TYPE	Modem type, Octet 5d
–>LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
–>LL_L2_PROTOCOL #ISO_1745 #Q.921 #X.25_LINK #X.25_MULTILINK #EXTENDED_LAPB #HDLC_ARM #HDLC_NRM #HDLC_ABM #LAN_LLC #X.75 #Q.922 #Q.922CORE #USERSPEC #T.90L2	Layer 2 Protocol Identifier, Octet 6 *Basic mode ISO 1745* *Recommendation Q.921 (i.441)* *Recommendation X.25, Link level* *Recommendation X.25 Multilink* *Extended LAPB; half duplex (T.71)* *HDLC ARM (ISO 4335)* *HDLC NRM (ISO 4335)* *HDLC ABM (ISO 4335)* *LAN Logical link control (ISO 8802/2)* *Rec. X.75, single link procedure* *Q.922* *Q.922 Core aspects* *User specified* *T.90 Layer 2 protocol*
–>LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
–>LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
–>LL_L3_PROTOCOL #Q.931 #X.25_PACKET #ISO_8208 #ISO_8348 #ISO_8473 #T.70 #USERSPEC #T.90L3	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer* *ISO 8208* *ISO 8348* *ISO 8473* *Rec. T.70, minimum network layer* *User specified* *T.90 Layer 3 protocol*
–>LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if –>LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if –>LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.1.31 HIGH LAYER COMPATIBILITY IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A

Selector Name and related Constants	Bellcore Reference
–>HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*Q.931 Standard*
#INTERNATIONAL	*Other International Standard*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Network specific*
–>HL_INTERPRETATION	Interpretation, Octet 3
#FIRST_CHARACTER	*First HL Characteristics to be Used*
–>HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
–>HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#MAINTENANCE	*Reserved for Maintenance (Note 1)*
#MANAGEMENT	*Reserved for Management (Note 1)*
#VIDEOTEL	*Videctelephony*
–>HL_CHARACTER_EXT	Extended High Layer Characteristics, Octet 4a
#TELEPHONY	*Telephony*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application*
#NOT_AVAIL1	*Not available for assignment*
#NOT_AVAIL2	*Not available for assignment*
#VIDEOTEL	*Videotelephony*

Note 1 : Only in this case, Octet 4a may follow

6.1.32 USER USER INFORMATION IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->UU_PROTOCOL_DISCR **#USER_SPECIFIC_PD** **#OSI_HIGHER_LAYER** **#X.244** **#SYS_MNG_CONVERG** **#IA5_CHAR** **#X.208** **#V.120** **#Q931_CALL_CONTROL**	Protocol discriminator, Octet 3 *User specific protocol* *OSI Higher Layer Protocols* *X.244* *System mng. convergence function* *IA5 characters* *X.208 / X.209* *V.120 rate adaption* *Q.931 Call Control Messages*
->UU_USER_INFO	User–User Information, Octet 4 *

6.2 CODESET 5 IE's

6.2.1 DISPLAY TEXT IE (I#DISPLAY_TEXT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->DT_DISPLAY_TYPE **#NORMAL**	Display type, Octet 3 *Normal*
->DT_DISPLAY_INFO	Display information, Octet 4 *

6.2.2 OPERATOR SYSTEM ACCESS IE (I#OPERATOR)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	Bellcore Reference
->OP_SERV_TYPE **#UNSPEC**	Type of service, Octet 3 *Unspecified*
->OP_ACCESS **#PUBL_PRINC** **#PUBL_ALT**	Type of access, Octet 3 *public / principal* *public / alternate*

6.2.3 SOFT KEY CONTROL IE (I#SOFTKEY)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_5

Selector Name and related Constants	Bellcore Reference
->SK_ACT **#DEACTIVARED** **#ACTIVATED**	Activate / Deactivate, Octet 3 *Deactivate* *Activate*
->SK_NUMBER	Soft Key Number, Octet 3
->SK_ID_NUMBER	Feature identifier number, Octet 4
->SK_ID_NUMBER_CONT	Feature identifier number, Octet 4a
->SK_DISPLAY_INFO	Display information, Octet 5 *

6.3 CODESET 6 IE's

6.3.1 CALL APPEARANCE IE (I#CALL_APP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A

Selector Name and related Constants	Bellcore Reference
->CA_ID	Call appearance identifier, Octet 3
->CA_ID_CONT	Call appearance identifier cont., Octet 3a

6.3.2 REDIRECTING SUBADDRESS IE (I#REDIRECTING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->RS_SPARE	Reserved, Octet 3
->RS_ADDRESS	Subaddress, Octet 4 *

6.3.3 REDIRECTION NUMBER IE (I#REDIRECTION_NUM6)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	Bellcore Reference
->RDIN6_NUMBER_PLAN_TYPE **#NATIONAL_PLAN** **#NETWORK_NUMBER** **#LOCAL_NUMBER** **#ABBREVIATED**	Type of Number and numbering Plan, Octet 3 *National number (E.164)* *Network specific number* *Local number (E.164)* *Abbreviated Number*
->RDIN6_NUMBER	Address Digits, Octet 4 *

X.25 LAYER 3
D-CHANNEL

User Manual for
Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	January, 1991

RELEASE NOTES R-1

1. GENERAL 1-1

1.1. INTRODUCTION 1-1

1.2. X.25 PACKET-LAYER 1-1

1.3. INTERFACE BETWEEN LAPD AND PACKET-LAYER 1-1

1.4. TESTMANAGER 1-2

1.5. RESTRICTIONS TO OTHER X.25 TESTPACKAGES RUNNING ON K1197 1-2

2. MENU SYSTEM 2-1

2.1. MOVING AND SELECTING ITEMS 2-1

2.2. MENU ICONS 2-2

2.3. FUNCTION KEYS 2-3

2.4. X.25 REPORT SELECTION 2-4
2.4.1. X.25 PACKET FILTER SELECTION 2-5
2.4.1.1. X.25 CHANNEL SELECTION 2-6
2.4.1.2. X.25 TRIGGER MENU 2-7
2.4.2. X.25 LAYER 3 TRIGGER SELECTION 2-8
2.4.2.1. X.25 REPORT CAPTURE/DISK CALL TRIGGER 2-10
2.4.2.1.1. X.25 LAYER 3 SETUP 2-11
2.4.2.1.2. X.25 LOGICAL LINKS 2-13
2.4.3. X.25 LOGICAL CHANNELS 2-14
2.4.3.1. X.25 LOGICAL CHANNEL SETUP 2-15
2.4.3.1.1. X.25 RECOMMENDATION SETUP 2-16

3. COMMANDS 3-1

3.1. SETUP COMMANDS 3-1
3.1.1. LAYER 3 SETUP COMMANDS 3-1
3.1.1.1. GENERAL SETUP COMMANDS 3-1
3.1.1.2. CHANNEL SPECIFIC SETUP COMMANDS 3-6

3.2. SEND PROTOCOL COMMANDS 3-8
3.2.1. LAYER 3 SEND COMMANDS 3-8

3.3. SEND PACKET COMMANDS 3-10

3.4. MISCELLANEOUS COMMANDS 3-14
3.4.1. PROTOCOL TIMERS 3-14
3.4.2. REFERENCE EMULATION STATE MACHINES 3-14
3.4.2.1. LAYER 3 STATE MACHINES 3-14

4. TEST MANAGER 4-1

4.1. EVENT RECOGNITION WORDS 4-1
4.1.1. ?RX_PACKET 4-1
4.1.2. ?RX 4-1
4.1.3. LCN-MULTIPLEXER 4-2

4.2. LIST OF COMMUNICATION VARIABLES 4-3

Program:	X.25 D-Channel
Software Version:	V3.1
Previous Version:	V3.0
System Version:	MF 2.2.
Revision Date:	January, 1991

☞ *Please note that it is important to read this paper before reading the manual, because some changed functions are described there.*

Enhancements in the User-Interface of the X.25 Application:

USER FACILITIES IN CALL REQUEST PACKET

USER FACILITIES IN CALL CONNECTED PACKET

USER DATA FIELD

REGISTRATION FIELD IN REGISTRATION PACKET

NEGOTIABLE PACKET AND WINDOW SIZE

NEGOTIABLE THROUGHPUT CLASS

SEND/ACCEPT REVERSE CHARGING

DIAGNOSTIC

USER SPECIFIED FACILITY NEGOTIATION

FAST SELECT

FAST SELECT WITH RESTRICTION

HUNT GROUP

CALL REDIRECTION SETUP

HEX EDITOR FOR FACILITIES

DATA PACKET ECHO

FAST SELECT FACILITY MODE

Enhancements of the X.25 Basis:

RESTART, CLEAR and RESET-INDICATION with 4 Byte only will be detected as to short by the DTE.

The length of the User-Data in CALL- and CLEAR-Packets will be computed

The length of the FACILITY-Field will be computed.

CALL USER DATA is enhanced to 128 Byte.

The default value of the PLAYBACK function is now PLAY=ON.

Layer 3 is enhanced to the function RUN_LAYER3.

1. GENERAL

1.1. INTRODUCTION

The X.31 Simulation/Emulation enables the user to send, receive and decode X.25 PDU's on a LAPD-Layer2. Further more, the Q930/Q931-Protocol is running in parallel and should be considered as the Base-Software for this package, because it is added with the X.25 layer3 protocol behaviour. That means, all the basic functions are available from the ISDN-Q931-Software.

The decision between Packet and Signalling data is done by checking the SAPI (values 16 or Ø). There is no special user-interface to support the X.25. All the features for X.25-Packet layer are available by several commands as known from K1197 - X.25 Software.

1.2. X.25 PACKET-LAYER

The packet layer is based on recommendation CCITT-80 or CCITT-84 (switchable). Full emulation of packet layer is supported. By switching off the emulation it is possible to simulate the protocol outside the state-machine. The Implementation contains of several commands (SETUP, send protocol, send packets, receive packets). They are described in the following chapters.

1.3. INTERFACE BETWEEN LAPD AND PACKET-LAYER

The interface between layer 2 and 3 has changed. From view of layer 3 (2) the appropriate service-primitives to layer 2 (3) are used. Moreover, up to 8 layer 2-links are supported. Every layer 2 link can carry up to 4 logical channels (including channel Ø for Restart). Normally it is useful to reserve some layer 2-links as signalling-links and some other as links for packet-data. (Note: Only links with preselected SAPI=16 are recognized as packet data links).

When using Layer3-send-commands on more than one Layer2-Link, it is useful to precede the send commands with the corresponding link selection to be on the right link.

If ISDN L3-Multiplexer is switched on, and a frame containing SAPI=16 is to be multiplexed, the CEI [CES + SAPI] and the LCN [logical channel number] are considered (Refer to ISDN Connection Management).
For packet data the LCN is used for multiplexing instead of the Call-Reference-value for signalling data.
Additionally, the channel structure, which carries the selected channel can be choosen by command, which forces the multiplexer to update the channel structure to the selected one. An example is given in chapter 4.1.3.
The Testscenarios (on the Test-scrip disk) give an example for using both, signalling and packet data, in parallel.

1.4. TESTMANAGER

Some Testmanager-recognition words with the appropride mnemonics for X.25 PDU's were added to the testpack. However the original Testmanager-concept as known from the ISDN-Software, is used.

1.5. RESTRICTIONS TO OTHER X.25 TESTPACKAGES RUNNING ON K1197

The number of logical channels per link is reduced to 4 (including CHØ for Restart).
Packet retransmisson is not supported.

Layer 2-LAPB is completely replaced by Layer 2-LAPD.

2. MENU SYSTEM

The menu system provides an intuitive interface designed to minimize the need to refer to the manual. It includes pop-up help screens to remind the user of meanings of selections available in the menus.

2.1. MOVING AND SELECTING ITEMS

Within a menu, the following keys may be used to move and perform actions:

Cursor up (↑)	Move to next selectable item above this one.
Cursor down (↓)	Move to next selectable item below this one.
Cursor left (←)	Move to previous menu to the left without entering current item. Note: Some menus do not allow this key in order to force the user to select some action through a function key before leaving the menu (e.g: the Hex editor and Message Building System). String and numeric entry fields: Move cursor one character backward.
Cursor right (→)	Move to next menu to the right if one exists. String and numeric entry fields: Move cursor one character forward (first key struck enables editing the already displayed string, which is indicated by a little blinking cursor).
Control cursor up (ctrl-↑)	Move up one page on scrolling menus.
Control cursor down (ctrl-↓)	Move down one page on scrolling menus.
Control shift cursor up (ctrl-shift-↑)	Move to first item in menu.
Control shift cursor down (ctrl-shift-↓)	Move to last item in menu.
Control shift cursor left (ctrl-shift-←)	Close all opened menu windows except the first one.
Control shift cursor right (ctrl-shift-→)	Open all those menu windows which were closed by ctrl-shift-← or a series of cursor left actions.
Shift cursor right (shift-→)	String and numeric entry fields: Shift the characters from this place on one step forward and create an empty place (blank) between the already existing characters.
Shift cursor left (shift-←)	String and numeric entry fields: Delete the character under the cursor and shift the following characters one step backward.
Control cursor right (ctrl-→)	String and numeric entry fields: Move the blinking cursor to the last character in the string.
Control cursor left (ctrl-←)	String and numeric entry fields: Move the blinking cursor to the first character in the string.
Home	Move to first item in menu (same as ctrl-shift-↑).
Return	Select an item, complete entry of a typed value or execute a command.

Back Space (BS), Delete (DEL)	String and numeric entry fields: Erase the character before the cursor.
Space bar	When the cursor is on a field which takes a number of options, the space bar may be used to scroll through the available options. These options may also be accessed by typing cursor right and selecting the desired option by pressing **Return**.
Number and character keys	Search a string from the list of selectable items. Every key struck moves the cursor forward onto the next item to which the string matches. String and numeric entry fields: Erase the old entry and create a new one. If cursor right, ctrl→, shft→ or shft-← was pressed first, edit the already existing string.

2.2. MENU ICONS

The menus use a number of icons to convey information about the options available on the menu. The borders of the menu are of two types: single line and double line. Double line borders are used to indicate the last menu in a chain. There will not be any menus to the right of a double bordered menu.

At the top and bottom of the right border of a menu there will apear arrows if the menu is bigger than what will fit in the window. The arrow indicates that there are more items outside the window in the direction indicated.

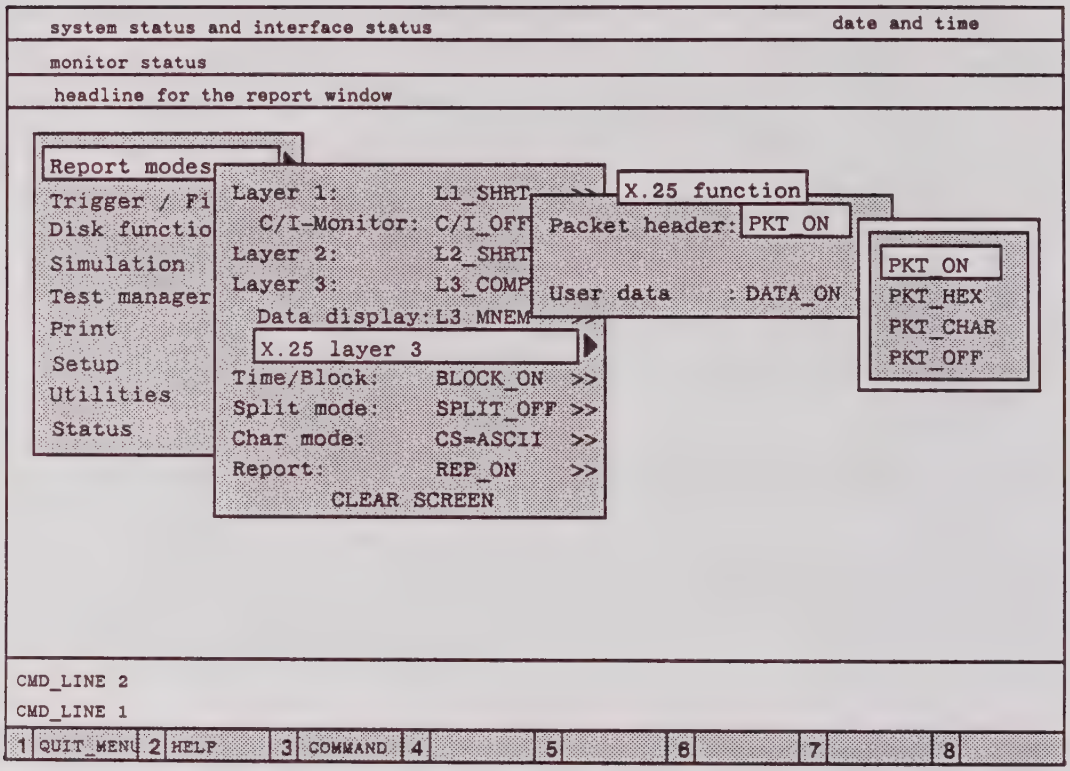
At the right of a selectable item, one of the following may appear: ▶ or ≫. The single solid arrow indicates that there is a menu to the right of this item that will be displayed when cursor right is typed. The double arrow indicates that the field takes a number of options which may be scrolled through with the space bar or listed using the cursor right key.

2.3. FUNCTION KEYS

A set of function keys on the bottom line of the display is almost always availble. Some common keys are listed here.

MENU / QUIT_MENU	: Enter to menu system control. / Leaves the menu system control.
HELP / QUIT_HELP	: Open a help menu for the particular menu window.
COMMAND / QUIT_COM	: Give the control to the FORTH interpreter which enables the use of keyboard for typing commands. The menu system window remains still visible. Return the control back to the menu system.
HALT	: Switch to HALT mode
FREEZE	: Switch to FREEZE mode
MONITOR	: Switch to MONITOR mode
TM_KEYS / MON_KEYS	: Switch to Test manager control keys / back to monitor control keys

2.4. X.25 REPORT SELECTION

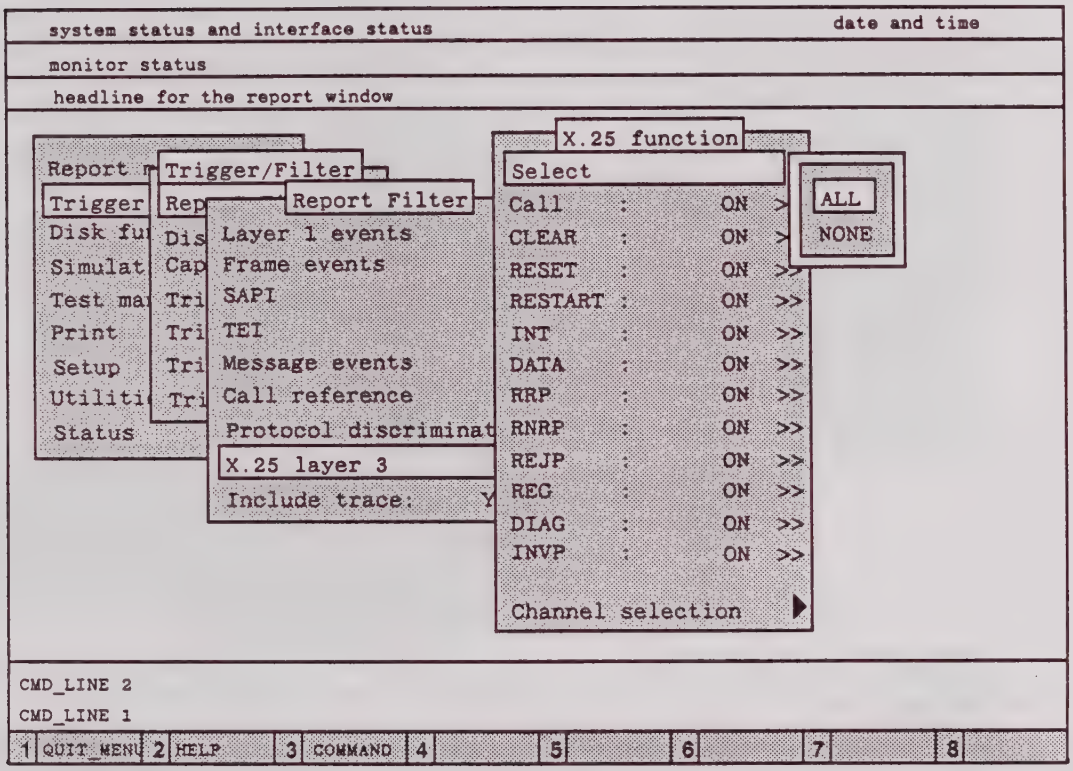


MENU SELECTION:

PACKET HEADER : Select complete packet report mode (ON/OFF/ HEX/CHAR)

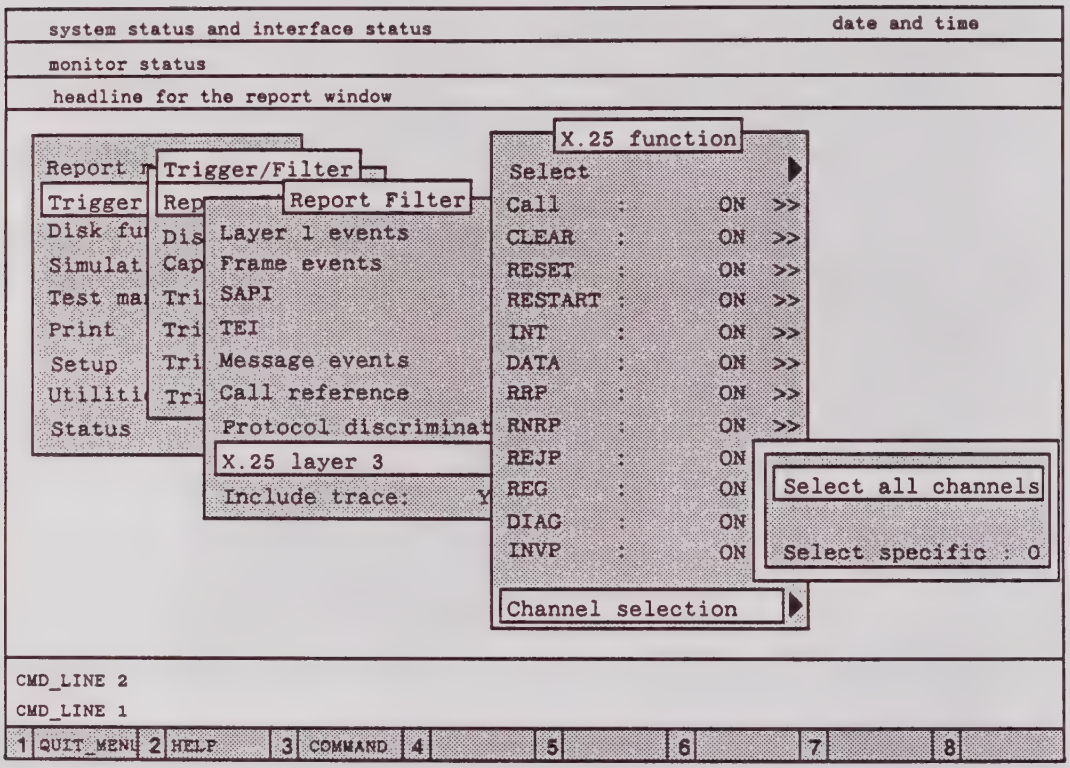
USER DATA : Select complete data report mode (ON/OFF/ HEX/CHAR)

2.4.1. X.25 PACKET FILTER SELECTION



MENU SELECTION:	
ALL	: Enable all layer 3 filter events
NONE	: Disable all layer 3 filter events
CALL	: Enable or disable filter for call
CLEAR	: Enable or disable filter for clear
RESET	: Enable or disable filter for reset
RESTART	: Enable or disable filter for restart
INT	: Enable or disable filter for interrupt
DATA	: Enable or disable filter for data
RRP	: Enable or disable filter for receive ready
RNRP	: Enable or disable filter for receive not ready
REJP	: Enable or disable filter for reject
REG	: Enable or disable filter for registration
DIAG	: Enable or disable filter for diagnostic
INVP	: Enable or disable filter for invalid packet
CHANNEL SELECTION	: Enter channel selection menu

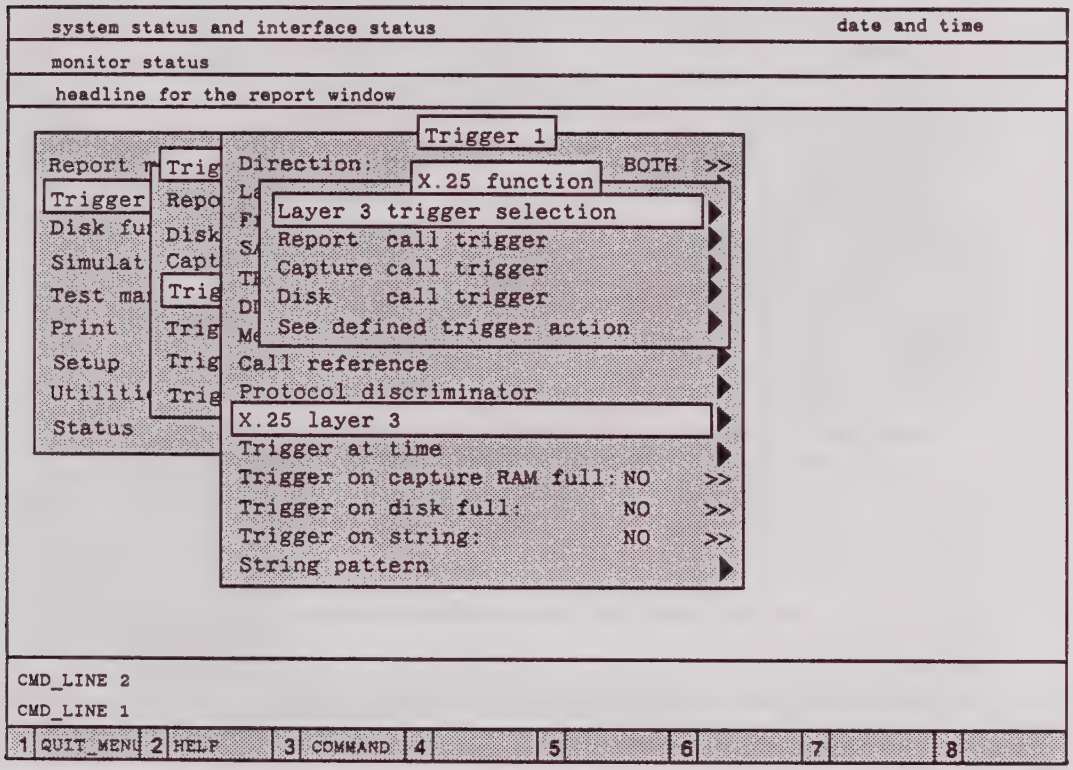
2.4.1.1. X.25 CHANNEL SELECTION



MENU SELECTION:

SELECT ALL CHANNELS	: Select al logical channels, wanted for report
SELECT SPECIFIC	: Enter specific logical channel, wanted for report

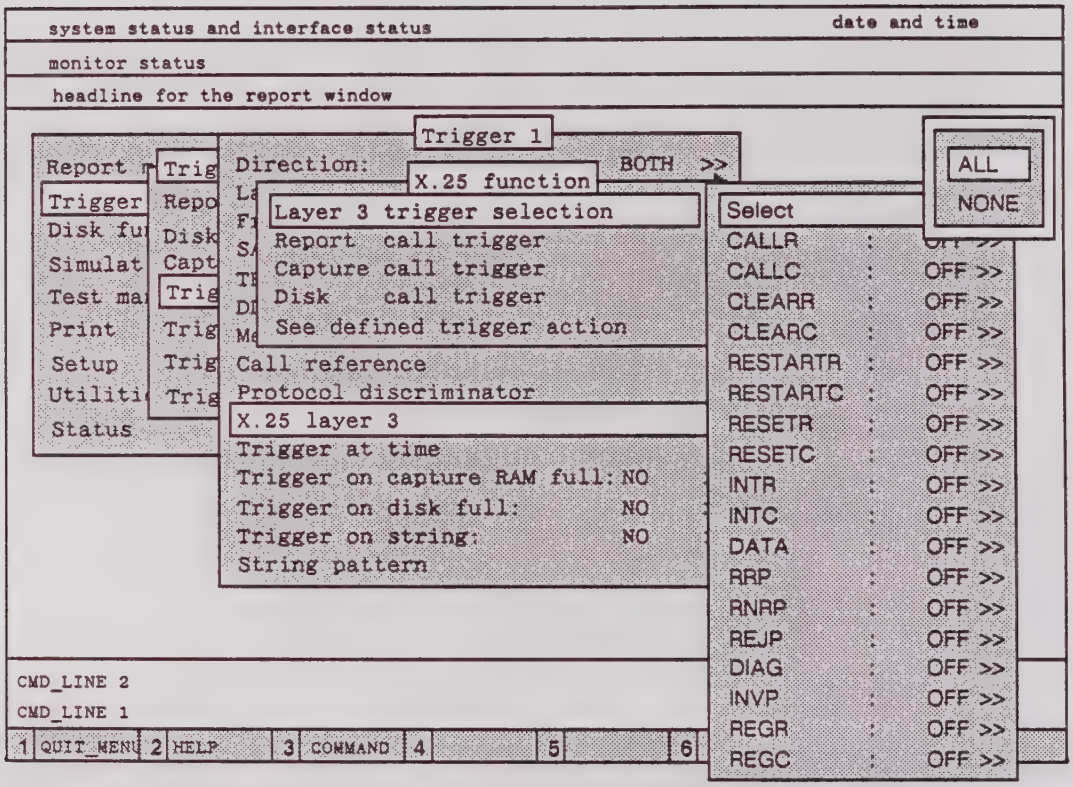
2.4.1.2. X.25 TRIGGER MENU



MENU SELECTION:

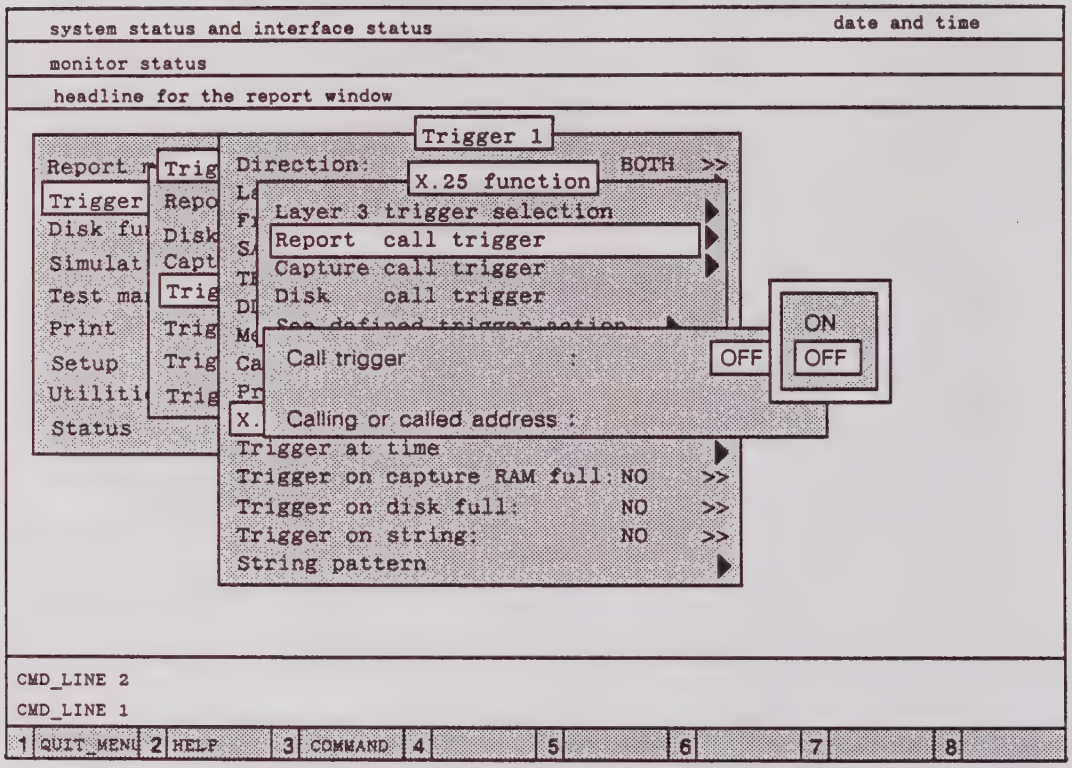
LAYER 3 TRIGGER SELECTION	: Enter menu to select trigger on layer 3 events
REPORT CALL TRIGGER	: Enter report call trigger selection menu
CAPTURE CALL TRIGGER	: Enter capture call trigger selection menu
DISK CALL TRIGGER	: Enter disk call trigger selection menu
SEE DEFINED TRIGGER ACTION	: Enter menu to see current trigger action

2.4.2. X.25 LAYER 3 TRIGGER SELECTION



MENU SELECTION:	
ALL	: Enable all layer 3 trigger events
NONE	: Disable all layer 3 trigger events
CALLR	: Enable or disable trigger for call request
CALLC	: Enable or disable trigger for call counected
CLEARR	: Enable or disable trigger for clear request
CLEARC	: Enable or disable trigger for clear confirmation
RESTARTR	: Enable or disable trigger for restart request
RESTARTC	: Enable or disable trigger for restart confirmation
RESETR	: Enable or disable trigger for reset request
RESETC	: Enable or disable trigger for reset confirmation
INTR	: Enable or disable trigger for interrupt
INTC	: Enable or disable trigger for interrupt confirmation
DATA	: Enable or disable trigger for data
RRP	: Enable or disable trigger for receive ready packet
RNRP	: Enable or disable trigger for receive not ready packet
REJP	: Enable or disable trigger for reject packet
DIAG	: Enable or disable trigger for diagnostic packet
INVP	: Enable or disable trigger for invalid packet
REGR	: Enable or disable trigger for registration request packet
REGC	: Enable or disable trigger for registration confirm packet

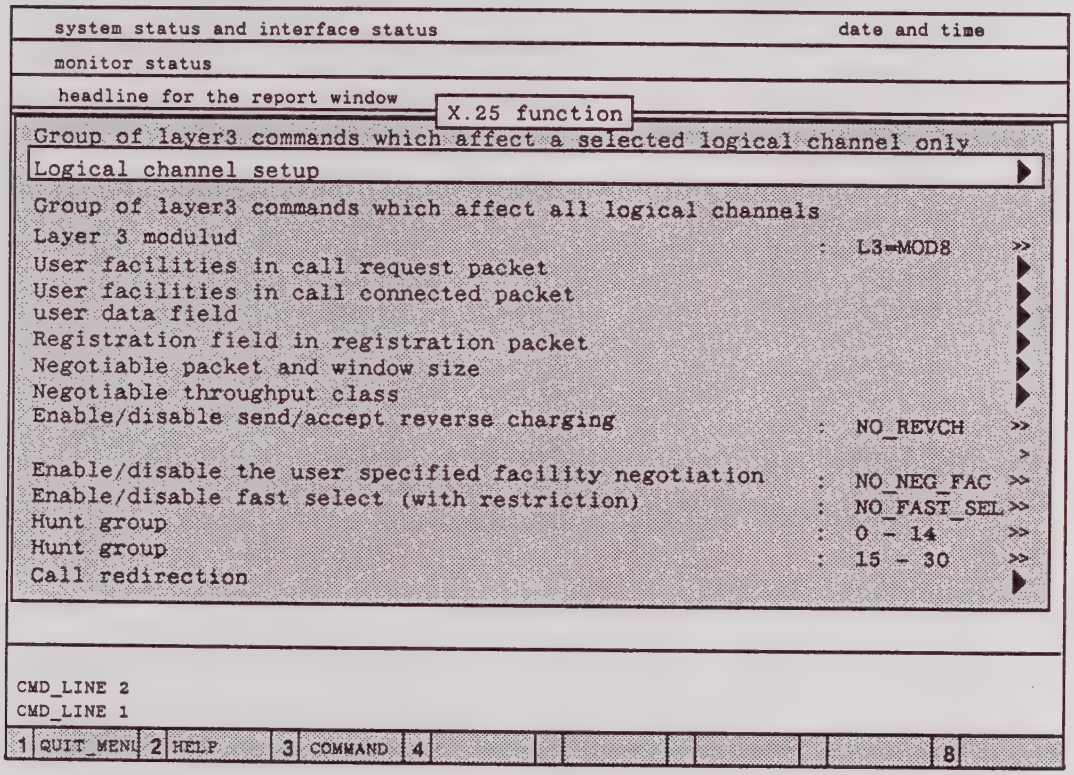
2.4.2.1. X.25 REPORT CAPTURE/DISK CALL TRIGGER



MENU SELECTION:

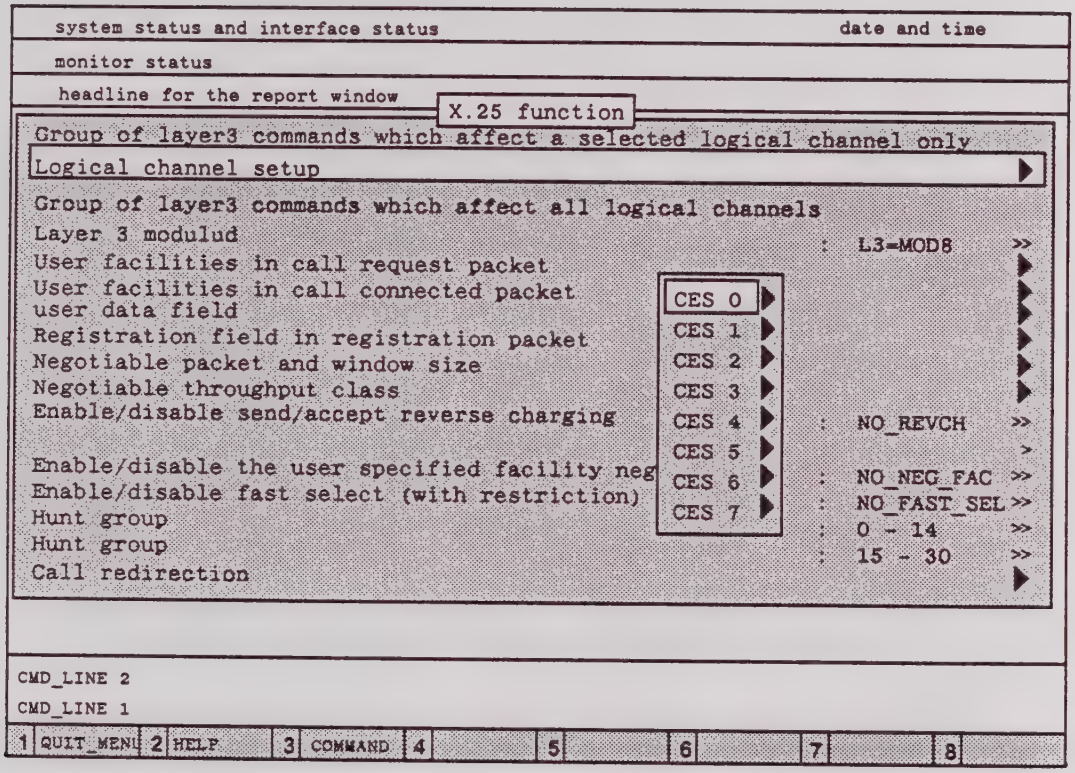
CALL TRIGGER	: Trigger on called/calling address ON or OFF
CALLING OR CALLED ADDRESS	: Enter calling or called address

2.4.2.1.1. X.25 LAYER 3 SETUP


system status and interface status
date and time
monitor status
headline for the report window
X.25 function
Group of layer3 commands which affect a selected logical channel only
Logical channel setup
Group of layer3 commands which affect all logical channels
Layer 3 modulud : L3=MOD8 >>
User facilities in call request packet
User facilities in call connected packet
user data field
Registration field in registration packet
Negotiable packet and window size
Negotiable throughput class
Enable/disable send/accept reverse charging : NO_REVCH >>
Enable/disable the user specified facility negotiation : NO_NEG_FAC >>
Enable/disable fast select (with restriction) : NO_FAST_SEL >>
Hunt group : 0 - 14 >>
Hunt group : 15 - 30 >>
Call redirection
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 8

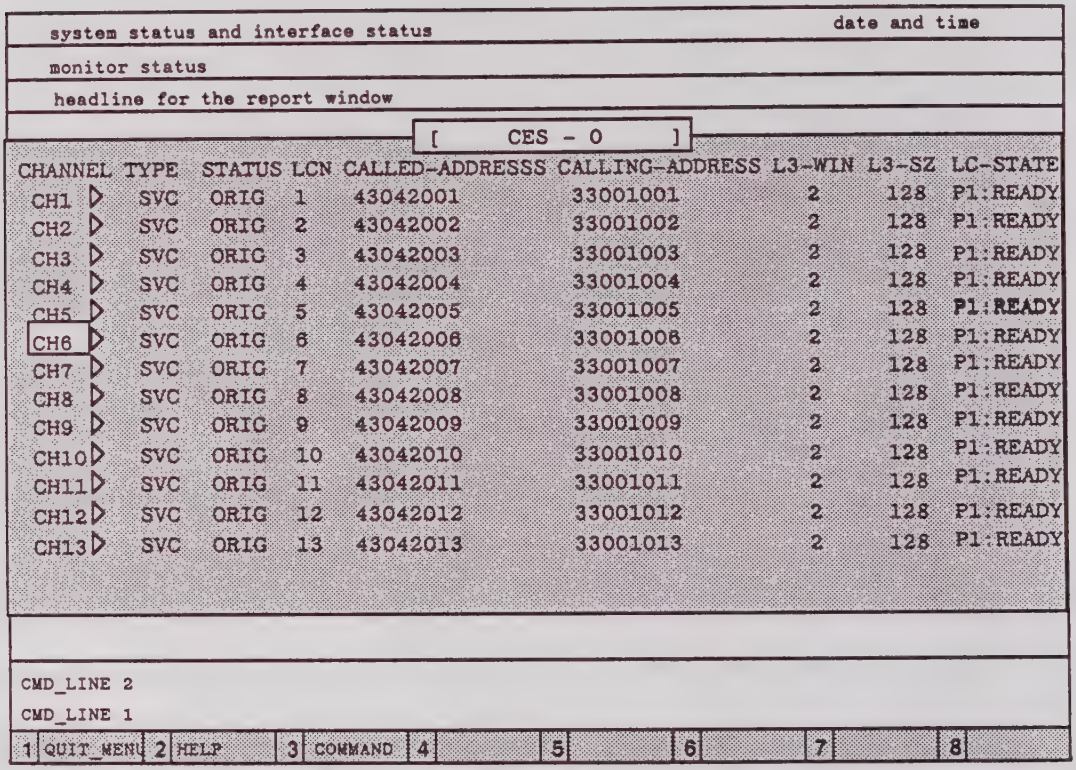
MENU SELECTION:	
Logical channel setup	: Enter logical link menu
Layer 3 modulus	: Select layer 3 modulus 8 or 128 mode
User facilities in call request packt	: Enter Menu User Facilities in Call Request Packet
User facilities in call connected packt	: Enter Menu User Facilities in Call Connected Packet
User data field	: Enter Menu User Data Field
Negotiable packet and window size	: Enter Menu Negotiable Packet and Window Size
Negotiable throuput class	: Enter Menu Negotiable Throughput Class
Send/access reverse charging	: Enable or disable Send/Access Reverse Charging
Diagnostic packets	: Enable or disable Diagnostic Packets
User specified facility negotiation	: Enable or disable User Specified Facility Negotiation
Fast select (with restriction)	: Enable or disable Fast Select
Hunt group	: Enter Menu Hunt Group
Call redirection setup	: Enter Menu Call Redirection

2.4.2.1.2. X.25 LOGICAL LINKS


system status and interface status
date and time
monitor status
headline for the report window
X.25 function
Group of layer3 commands which affect a selected logical channel only
Logical channel setup
Group of layer3 commands which affect all logical channels
Layer 3 modulud : L3=MOD8 >>
User facilities in call request packet
User facilities in call connected packet
user data field
Registration field in registration packet
Negotiable packet and window size
Negotiable throughput class
Enable/disable send/accept reverse charging : NO_REVCH >>
Enable/disable the user specified facility neg : NO_NEG_FAC >>
Enable/disable fast select (with restriction) : NO_FAST_SEL >>
Hunt group : 0 - 14 >>
Hunt group : 15 - 30 >>
Call redirection
CES 0
CES 1
CES 2
CES 3
CES 4
CES 5
CES 6
CES 7
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:	
CES0/.../CES7	: Enter logical channel tabe for logical link (0...7)

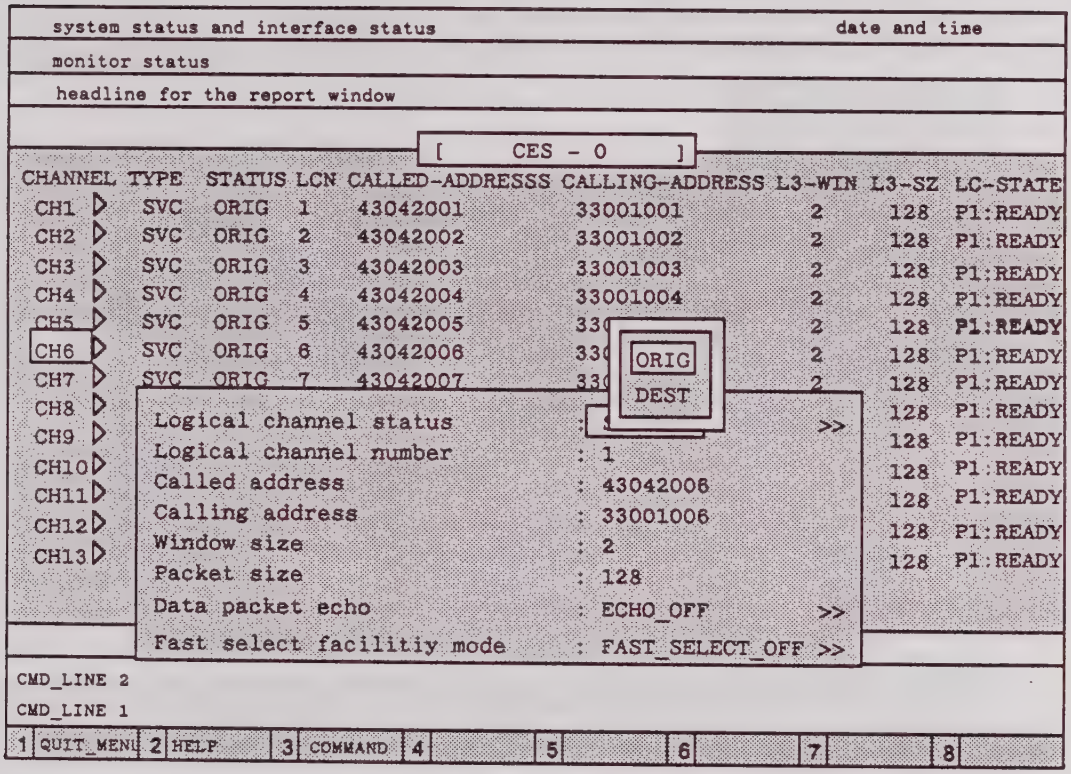
2.4.3. X.25 LOGICAL CHANNELS


system status and interface status
date and time
monitor status
headline for the report window
[CES - 0]
CHANNEL TYPE STATUS LCN CALLED-ADDRESSS CALLING-ADDRESS L3-WIN L3-SZ LC-STATE
CH1 SVC ORIG 1 43042001 33001001 2 128 P1:READY
CH2 SVC ORIG 2 43042002 33001002 2 128 P1:READY
CH3 SVC ORIG 3 43042003 33001003 2 128 P1:READY
CH4 SVC ORIG 4 43042004 33001004 2 128 P1:READY
CH5 SVC ORIG 5 43042005 33001005 2 128 P1:READY
CH6 SVC ORIG 6 43042006 33001006 2 128 P1:READY
CH7 SVC ORIG 7 43042007 33001007 2 128 P1:READY
CH8 SVC ORIG 8 43042008 33001008 2 128 P1:READY
CH9 SVC ORIG 9 43042009 33001009 2 128 P1:READY
CH10 SVC ORIG 10 43042010 33001010 2 128 P1:READY
CH11 SVC ORIG 11 43042011 33001011 2 128 P1:READY
CH12 SVC ORIG 12 43042012 33001012 2 128 P1:READY
CH13 SVC ORIG 13 43042013 33001013 2 128 P1:READY
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:

CH1/.../CH3	: Enter menu logical channel setup

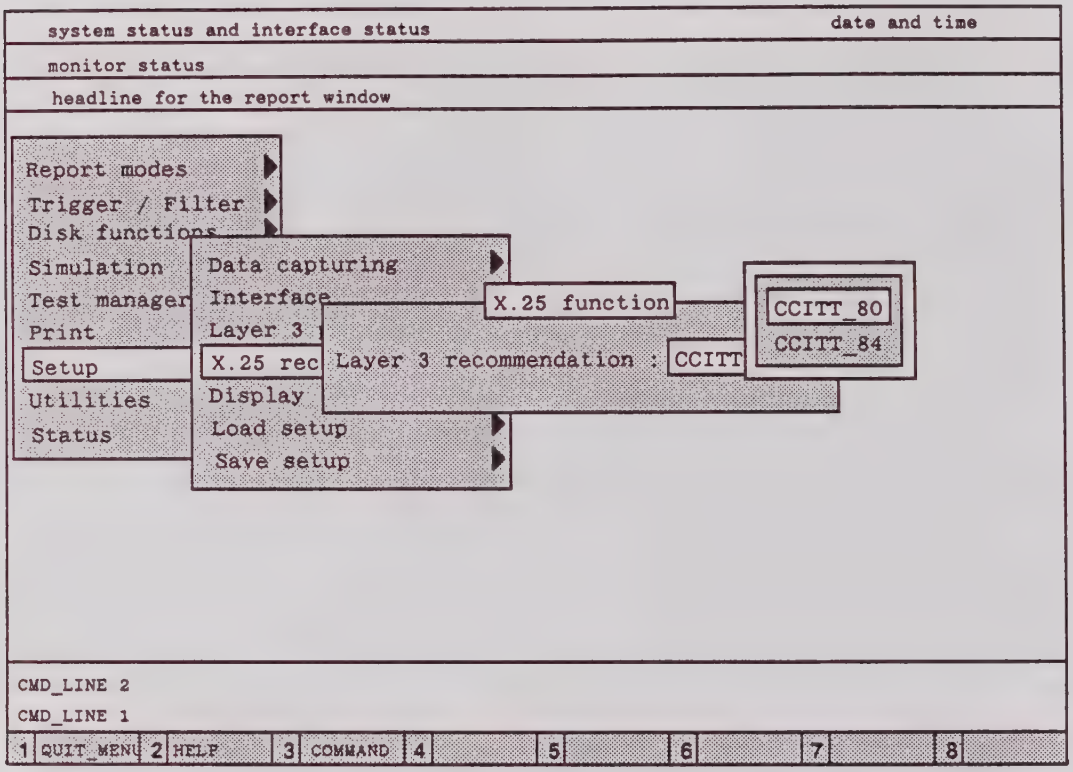
2.4.3.1. X.25 LOGICAL CHANNEL SETUP


system status and interface status
date and time
monitor status
headline for the report window
[CES - 0]
CHANNEL TYPE STATUS LCN CALLED-ADDRESSS CALLING-ADDRESS L3-WIN L3-SZ LC-STATE
CH1 SVC ORIG 1 43042001 33001001 2 128 P1:READY
CH2 SVC ORIG 2 43042002 33001002 2 128 P1:READY
CH3 SVC ORIG 3 43042003 33001003 2 128 P1:READY
CH4 SVC ORIG 4 43042004 33001004 2 128 P1:READY
CH5 SVC ORIG 5 43042005 33 2 128 P1:READY
CH6 SVC ORIG 6 43042006 33 2 128 P1:READY
CH7 SVC ORIG 7 43042007 33 2 128 P1:READY
CH8 128 P1:READY
CH9 128 P1:READY
CH10 128 P1:READY
CH11 128 P1:READY
CH12 128 P1:READY
CH13 128 P1:READY
ORIG
DEST
Logical channel status : >>
Logical channel number : 1
Called address : 43042006
Calling address : 33001006
Window size : 2
Packet size : 128
Data packet echo : ECHO_OFF >>
Fast select facilitiy mode : FAST_SELECT_OFF >>
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:

Logical channel status	: Select originate or destination call
Logical channel number	: Enter logical channel value (valid 1 to 4095)
Called address	: Enter called address
Calling address	: Enter calling address
Window size	: Enter packet window size per logical channel
Packet size	: Enter data packet size per logical channel
Data packet echo	: Enable or disable Data Packet Echo
Fast select facility mode	: Select Fast Select Facility Mode

2.4.3.1.1. X.25 RECOMMENDATION SETUP



MENU SELECTION:

Layer 3 recommendation : Select recommendation 80 or 84

3. COMMANDS

This section describes the commands that are used to setup the emulation and to control the protocol operations. These commands are classified into the following categories: SETUP commands, SEND PROTOCOL commands, SEND PACKET commands and Miscellaneous commands. These commands can be either entered directly to the command line or used in the Test Manager scenario contexts.

3.1. SETUP COMMANDS

This section lists the commands for setting up the various user options associated with the protocol.

Layer 3 setup commands concern with the modification of variables and system parameters in Layer 3, and the selection of network utilities in the logical channel. By using this group of commands, the user can change the modulo and the values of the timers used in Layer 3. Furthermore, the user can disable or enable the use of network utility field and/or the user facility field. The user can set the various network utility options, such as the throughput class, the window size, and the packet size. The user can also create his own network utility or user facility field to be used in the virtual call.

3.1.1. LAYER 3 SETUP COMMANDS

The Layer 3 setup commands are divided into two groups. The first group of commands, when issued, will affect all logical channels. The second group of commands will have effect only on the selected logical channels.

3.1.1.1. GENERAL SETUP COMMANDS

CLASS (—)
Set the throughput class facility. The valid values are 75, 150, 300, 600, 1200, 2400, 4800, 9600, 19200 and 48000 (bit/s) according to the Rec. X.25. The default value is set to 9600.
Usage: CLASS 4800

NO_FAC (—)
No user facility field will appear in the Call Request packet. This is the default mode. The only mode for using the user facility field in Call Request packet is by the command USER_FAC which will put the user defined facility field in the Call Request packet.

YES_FAC (—)
Facility negotiation will be used when Call Request Packets are sent. Data packet size, packet window, throughput class and fast select are negotiated. (Automatic default facility)

USER_FAC (—)
This command instructs the emulation to include the user facility field defined by the user with the MAKE_FAC command in the Call Request packet. This command remains in effect until a NO_FAC is issued.

CCITT-80 (—)
This command forces the automatic emulation to satify the Recommendation Version 1980 [3] (Yellow Book).

CCITT-84 (—)
This command forces the automatic emulation to satify the Recommendation Version 1984 [4] (Red Book).

MAKE_FAC (string --)

Define the facility field to be used in Call Requests when USER_FAC is selected. The input is in HEX and a space is needed between X" and the input. The first byte in the input is the user facility length with the following bytes to be used in the user facility field. The input should be limited to 64 bytes by CCITT 80 or 109 bytes by CCITT 84 (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional cases. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the facility field in the same order as they are entered.

Example: `X" 03420808" MAKE_FAC`

This will define user facility for packet size of 256 bytes.

NO_CA (--)

No address field in the Call Connected packet.

YES_CA (--)

The address field received in the Call Request packet will be used in the Call Connected packet. This is the default mode.

NO_CAFAC (--)

No user facility field in the Call Connected packet. There are two modes to use the user facility field in the Call Connected Packets:

(i) **ECHO_CAFAC** will echo the user facility field received in the Call Request packet. This is the default mode.

(ii) **USER_CAFAC** will have user defined user facility fields appear in Call Connected packets.

This command will remain in effect until ECHO_CAFAC or USER_CAFAC is issued.

ECHO_CAFAC (--)

Echo the facility field received in Call Request packet in the Call Connected packet. This is the default setting.

USER_CAFAC (--)

Instruct the emulation to include the user facility field defined by the user using the MAKE_CAFAC command in the Call Connected or Call Request packet. This command remains in effect until NO_CAFAC or ECHO_CAFAC is issued.

MAKE_CAFAC (string --)

Define the user facility field to be used in Call Connected packet when USER_CAFAC is selected. The first byte in the input is the user facility length with the following bytes to be used in the user facility field. The input should be limited to 64 bytes (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional conditions. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the facility field in the same order as they are entered.

Example: `X" 03420808" MAKE_CAFAC`

This will define user facility for packet size of 256 bytes.

MAKE_CUD **(string --)**

Make user data field for the next Call Request, Call Connected or Clear Request packet to be sent. There are two usages:

(i) ASCII data **Usage:** `" text input" MAKE_CUD`

(ii) HEX data **Usage:** `X" hex inpuut" MAKE_CUD`

The data field may be up to 16 or 128 octets depending if the Fast Select optional user facility is used. It is the user responsibility to make sure this restriction is being satisfied. The user is allowed to construct a user data field of up to 240 bytes to generate exceptional cases. Inputs that are longer than 240 bytes will be truncated with only the first 240 bytes being used. The input will appear in the user data field in the same order as they are entered, i.e. the first entry will go into the first octet in the user data field. This command remains in effect until 0 MAKE_CUD is issued.

Example: `" SIEMENS K1195" MAKE_CUD`

Note: The user data field for Call Connected or Clear Request packet is sent only if the Fast Select network utility option has been selected.

NO_REG **(--)**

No registration field will appear in the Registration packets. This is the default mode. The only mode for using the registration field in Registration packets is by the command USER_REG which will put the user defined registration field in the Registration packets. This is the default mode.

USER_REG **(--)**

This command instructs the emulation to include the registration field defined by the user with the MAKE_REG command in the Registration packets. This command remains in effect until
a NO_REG is issued.

MAKE_REG **(string--)**

Define the registration field to be used in a Registration packet when USER_REG is selected. The input is in HEX and a space is needed between X" and the input. The first byte in the input is the registration length with the following bytes to be used in the registration field. The input should be limited to 64 bytes (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional cases. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the registration field in the same order as they are entered.

Example: X" 03420707" MAKE_REG

This will define registration for packet size of 128 bytes.

YES_REGADR **(--)**

Address field will be sent in the Registration packets.

NO_REGADR **(--)**

No address field in the Registration packets. This is the default mode.

The following commands allow the user to set the values for several facilities which will be negotiated in the Call Setup procedure. These facilities are packet- and window size as well as throughput class. Also the user may specify whether or not the facilities reverse charging or fast select will be accepted.
These user specified facilities can be sent within a normally sent Call request packet. If the Call accepted packet following contains values for the facilities these will be accepted, if they are less than or equal to the values sent in the Call request packet.
On the other hand received Call request packets will be accepted if they contain facilities which are specified by the user. Otherwise the Call will be rejected by a Clear request. If the Call received contains values higher than specified, a Call accepted is sent with facility values equal to the specified ones.

FLOW_CONTROL_NEG **(value packet size/ value window size --)**
Set the negotiable packet and window size values.

NO_FLOW_NEG **(--)**
Do not negotiate flow control.

CLASS_NEG **(value --)**
Set the negotiable throughput class value.

NO_CLASS_NEG **(--)**
Do not negotiate throughput class.

YES_REVCH **(--)**
Send/accept reverse charging.

NO_REVCH **(--)**
Do not send/accept reverse charging.

YES_FAST_SEL **(--)**
Send/accept fast select.

NO_FAST_SEL **(--)**
Do not send/accept fast select.

FAST_SEL_RE **(--)**
Send/accept fast select with restriction.

YES_NEG_FAC **(--)**
The user specified facilities will be negotiated.

NO_NEG_FAC **(--)**
The user specified facilities will not be negotiated.

SAVE_FAC **("filename" --)**
Saves the user specified facilities (including values) to disk.

LOAD_FAC **("filename" --)**
To load the user specified facilities saved with the command SAVE_FAC.

This X.25 128 Test Pack gives the user the ability to use basic (mod 8) or extended (mode 128) sequence numbering for the Packet layer.

The following commands are available to choose the sequence numbering for Layer 3 (common to all logical channels) :

L3_MODULO8 **(--)**
This command will set the Layer 3 into modulo 8 mode. The following packets will be sent with a General Format Identifier (GFI) of 0x01.

L3_MODULO128 **(--)**
This command will set the Layer 3 into modulo 128 mode. The following packets will be sent with a General Format Identifier (GFI) of 0x02.

L3_MODULUS **(-- n)**
This commans leaves on the stack the Packet Layer modulus value (8 or 128).

A received packet with an unexpected sequence format will be treated as a GFI error by the emulation. If the monitor is being used with no emulation, it is not necessary to set the modulo with the above FORTH words, as each received packet is decoded according to its GFI bits.

The window sizes for the logical channels can be set in the usual way to any value less than the modulus. The same restrictions as explained for the Link layer apply to the Packet layer, too.

The following commands allow the emulation to simulate a Hunt Group or Call Redirection respectively:

YES_HUNT/ NO_HUNT **(--)**
The facility Hunt Group is/is not subscribed to.

=HUNT_ADDRESS **(adr/busy-flag/n --)**
This command allows the user to set the hunt group addresses.

n=0	indicates the hunt group address
n=1-30	indicate the hunt group member addresses
busy-flag	(#BUSY or #FREE) sets the addresses busy or free (no effect for n=0)
adr	is a packed string containing the addresses

Usage: `" 1537489" #FREE 1 =HUNT_ADDRESS`
sets the address of hunt group member no. 1 to 1537489 and indicates that this member is able to accept a call.
If a Call request is received with the called address equal to the address of the hunt group (n=0) the next free hunt group member address will be sent in the Call accepted with facility "Called line address modified notification". If no address is available (free) a Clear request is sent. The algorithm to find out the next free address works "round robin".

HUNT_FREE / HUNT_BUSY **(n --)**
sets hunt group member n in free/busy mode.

YES_REDIRECTION / NO_REDIRECTION **(--)**
The facility Call Redirection is/is not subscribed to.

=REDIR_ADDRESS **(adr/busy-flag/n --)**

This command allows the user to set the Call redirection addresses.

n=0	indicates the original address
n=1-7	indicate the alternate addresses
busy-flag	(#FREE or #BUSY) sets the addresses busy or free
adr	is a packed string containing the address

Usage: `" 597348" #BUSY 0 =REDIR_ADDRESS`
sets the original address to 597348 and indicates that this number is busy.

If a Call request is received with the called address equal to the original address to be redirected and this number is busy the next free alternate address will be sent in the Call accepted with facility "Called line address modified notification". If no address is available (free) a Clear request is sent. The algorithm to find out the next free address always starts with address no. 1.

REDIR_FREE /

REDIR_BUSY **(n --)**
sets redirection address no. n in free/busy mode.

3.1.1.2. CHANNEL SPECIFIC SETUP COMMANDS

CH1 **(--)**
Selects CH1 as the current logical channel. Once the current logical channel is selected, the various parameters associated with the selected channels can be set by using the following sequence of commands:

Example: `CH1 SVC ORIG 1 44003300 55005500`

CH1	Selects CH1 logical channel.
SVC	Sets the logical channel to be a switched virtual circuit.
ORIG (or DEST)	Sets the logical channel to be used to originate calls or to be a destination. The logical channel number, the called and calling address must be provided after the command ORIG, and the logical channel number must be provided following the command DEST.
1	The logical channel value is entered in decimal as a combined logical channel and group value. Valid numbers are 1 to 4095. If the specified value is greater than 4095, 4095 will be used.
44003300	The called DTE's address field for this logical channel. The maximum address length is 15 decimal digits. (Not required if DEST is used). If length is greater than 15, only the first 15 digits will be used.
55005500	The calling address field for this logical channel. The maximum address length is 15 decimal digits. (Not required if DEST is used). If length is greater than 15, only the first 15 digits will be used.

=CALLED **(string --)**
The command =CALLED will change the called adress field used in call request packets. Use `0 LCNCALLED !` to get a zero length called address field.

Usage: `" address" =CALLED`

=CALLING **(string --)**

The command =CALLING will change the calling address field used in call request packets. Use 0 LCNCALLING ! to get a zero length calling address field.

Usage: " address" =CALLING

LCN **(-- addr)**

The word LCN provides a pointer to the logical channel value for the currently selected logical channel. Thus, any 12 bit value can be stored and will be used as the logical group and channel value.

CH2 / .. / CH3 **(--)**

Functions as the CH1 command for selecting one of the 3 logical channels available.

ECHO_ON **(--)**

All data received in data packets will be transmitted as a data packet on the same logical channel. This command will remain in effect until **ECHO_OFF** is entered.

All T.70 data packets received will be echoed back to the sender. This command will remain in effect until ECHO_OFF is specified.

ECHO_OFF **(--)**

Disable data packet echo. The ECHO_ON and ECHO_OFF commands affect only the selected logical channel (CH1 to CH3). ECHO_OFF is the default setting.

FAST_SEL_OFF **(--)**

The Fast Select facility is not to be used on the specified logical channel. This is the default mode for all logical channels.

Example: CH1 FAST_SELECT_OFF

FAST_SEL_ON **(--)**

The Fast Select facility with no restriction is to be used for the specified logical channel. When FAST_SELECT_ON is active, Clear Request packets will contain calling and called address fields, facility fields and clear user data. The clear confirm packets will contain calling and called address fields and facilities.

Example: CH2 FAST_SELECT_ON

FAST_SEL_RES **(--)**

The Fast Select facility with restriction is to be used for the specified logical channel. When FAST_SELECT_RESTRICTION is active and a Call Request package is received, a Clear Request packet with address, facility and user data fields will be generated.

Example: CH3 FAST_SELECT_RESTRICTION

SET_WINDOW **(value --)**

Set the packet window size per logical channel for sequence numbering used in Layer 3 flow control. Although the user can set the window size to any value, the effective window size achievable is limited by the amount of buffers available, which is in turn determined by the maximum frame length. The default value is set to 2.

Usage: CH1 7 SET_WINDOW

SET_SIZE ({2...2048} --)
Set the data packet size per logical channel. The valid values are from 2 to 1024 bytes. The default value is 128 bytes.
Usage: `CH1 256 SET_SIZE`
Note: The user should be aware of the fact that the frame size is required to be large enough to accommodate the new data packet size.

3.2. SEND PROTOCOL COMMANDS

The SEND PROTOCOL commands use the state machines of the emulation to control the protocol procedure. (See Section 3.3 for a description of the reference emulation state machines). These commands will force protocol state changes and the emulation will expect the correct response. However, only those commands that are appropriate to the current protocol state will be executed; invalid or unexpected commands will be ignored, and no action will be taken. The S: group of send frame and packet commands should be used to generate frames or packets without using the state machines of the emulation. The state machines of the emulation must be activated prior to the use of this group of commands. (Refer to Section 3.3 for commands that turn the state machines ON or OFF).

3.2.1. LAYER 3 SEND COMMANDS

RESTART (--)
A Restart Request packet will be generated and sent. This will clear all the logical channels. The Layer 3 state machines will enter state 3 (WAITING RESTART CONFIRM) after the packet is sent.

CALL (--)
A Call Request packet will be generated and sent on the selected logical channel. The called address and calling address fields are setup with the CHx commands. The Layer 3 state machines will enter LCN state 2 (P2 CALLING).

CALLS (value --)
Send a Call Request packet on several logical channels.

Usage: `x CALLS`

A Call Request packet is sent on each logical channel starting at LCN 1 up to LCN x. All these logical channels will enter LCN state 2 (P2 CALLING) after the execution of the command.

RESET (--)
A Reset Request packet will be generated and sent using the selected logical channel. The Layer 3 state machines will enter LCN state 5 (D2 RESET).

CLEAR (--)
A Clear Request packet will be generated and sent using the selected logical channel. The Layer 3 state machines will enter LCN state 6 (P6 CLEARING).

CLEARS (value --)
Send a Clear Request packet on several logical channels.

Usage: `x CLEARS`

A Clear Request packet is sent on each logical channel starting at LCN 1 up to LCN x. All these logical channels will enter LCN state 6 (P6 CLEARING) after the execution of this command.

INTERRUPT **(--)**

An Interrupt Request packet is sent on the selected logical channel. This logical channel must already have a call setup.

DATA **(--)**

A data packet is sent on the selected logical channel using data buffer 0 as the user data. This logical channel must already have a call setup before using this command. Data buffer 0 can support up to 1024 bytes of data.

Note: 1. The data packets are sent only if the packet window is open prior to processing the data packet. If the window is closed, the command is ignored.

2. The term 'selected logical channel' refers to the logical channel which has been activated by the last CHx command or the last received logical channel specific packet (eg. Call Connect packet), received data packet, or received flow control packet.

SENDP **(string --)**

Send a user defined packet. There are two usages:

(i) ASCII data **Usage:** `" data" SENDP`

(ii) HEX data **Usage:** `X" data" SENDP`

A maximum of 80* bytes can be used by keyboard. For input that is longer that this length, only the first 80 bytes will be used. A maximum of 80 bytes can be entered. For input that is longer than this length, only the first 80 bytes are used. The defined packet will be transmitted in an information frame with the correct frame sequence values.

SENDD **(string --)**

Send a user defined Data field in a data packet. There are two usages:

(i) ASCII data **Usage:** `" data" SENDD`

(ii) HEX data **Usage:** `X" data" SENDD`

A maximum of 80* bytes of can be used by keyboard. For input that is longer than this length, only the first 80 bytes are used. The data will be transmitted in an data packet with the correct packet sequence values.

Note: When using these commands and more than one Layer2-Link is established, use Link-selection to be on the right link.

example: `3 CES RESTART` `(Restart is sent on the Link which is selected by the CES=3)`

* For sending Data fields longer than 80 Byte the user has to edit a file which includes the wanted string followed by the appropriate commmand. The maximum string length is 256. After editing thr file has to be executed with the EXEC-Command. The defined packet will have been sent after compiling of the EXEC Command.

3.3. SEND PACKET COMMANDS

The S: send packet commands allow for sending packets at any time without regard to correct protocol procedure. These commands will not update the protocol states, or increment sequence values. They are designed to be used in test scenarios when frames and packets are to be generated out of context.

The send packet commands can be used alone if the Layer 2 state machine is in the information transfer states (states 5 to 8). If the Layer 2 state machine is not in the information transfer states, each send packet command should be followed by the S:INF command to send the I frames which contain the transmitted packets. Each I frame can contain, at most, one packet at a time.

S:RESTARTR (--)

Send a Restart Request packet on logical channel CH0 using values from the following variables:

- LCN – Logical group and channel value
- SENDGFI – GFI modulo value (1 or 2)
- SCAUSE – Cause value
- SDIAG – Diagnostic value

S:RESTARTC (--)

Send a Restart Confirm packet on logical channel CH0 using values from the following variables:

- LCN – Logical group and channel value
- SENDGFI – GFI modulo value (1 or 2)

S:CALLR (--)

Send a Call Request/Incoming Call packet on selected channel using values from the following variables:

- LCN – Logical group and channel value
- SENDDB – D bit value
- SENDGFI – GFI modulo value (1 or 2)
- LCNCALLED – Called address field with the first byte containing the length and with the address bytes in ASCII
- LCNCALLING – Calling address field with the first byte containing the length and with the address bytes in ASCII
- FACILITY-ACTIVE – Indication of type of facility negotiation desired
  - 0 = Facilities not wanted
  - 1 = Negotiate facilities for size, window, an throughput class
  - 2 = Send facilities defined with MAKE_FAC
- CUD-BUFFER – Call user data field buffer. The first byte is the length of the call user data field

S:CALLC (--)

Send a Call Connected/Accepted Call packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDDB	– D bit value
SENDGFI	– GFI modulo value (1 or 2)
RCALLED	– Called address field with the first byte containing the length and with the address bytes in ASCII
RCALLING	– Calling address field with the first byte containing the length and with the address bytes in ASCII
CAFAC-ACTIVE	– Indication of type of facility negotiation desired 0 = NO_CAFAC – do not send facilities 1 = ECHO_CAFAC – echo received facilities from buffer RFAC 2 = USER_CAFAC – Send facilities defined by MAKE_CAFAC

S:CLEARR (--)

Send a Clear Request/Indication packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SCAUSE	– Cause value
SDIAG	– Diagnostic value

When FAST select facility is active, the following are used for Clear Request Packets.

LCNCALLED	– Called address field
LCNCALLING	– Calling address field
FACILITY-ACTIVE	– Indication of type of facility negotiation desired
CUD-BUFFER	– Call user data field buffer. The first byte is the length of the call user data field

S:CLEARC (--)

Send a Clear Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:RESETR (--)

Send a Reset Request/Indication packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SCAUSE	– Cause value
SDIAG	– Diagnostic value

S:RESETC (--)

Send a Reset Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:INTR (--)

Send an Interrupt packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SIUD	– Interrupt user data byte

S:INTC (—)

Send an Interrupt Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:DATAP (—)

Send a Data packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDQB	– Q bit value
SENDDB	– D bit value
SENDGFI	– GFI modulo value (1 or 2)
NVS	– Ps value
NVR	– Pr value
SENDM	– More bit value
DATA-BUFFER	– Buffer containing 256 bytes of text
LCNSIZE	– Length of the data field to send

S:RRP (—)

Send a RR packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
NVR	– Pr value

S:RNRP (—)

Send a RNR packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
NVR	– Pr value

S:DIAG (—)

Send a DIAG packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SDIAG	– Diagnostic value
SDIAG-EXP	– Diagnostic explanation field. The first byte is the length of the field followed by up to 3 bytes for the explanation field to be sent

S:REJP (--)

Send a REJP packet on selected channel using values from the following variables:

LCN	- Logical group and channel value
SENDGFI	- GFI modulo value (1 or 2)
NVR	- Pr value

S:REGISTR (--)

Send a Registration Request packet on selected channel using values from the following variables:

LCN	Logical group and channel value
SENDQB	Q bit value
SENDDB	D bit value
SENDGFI	GFI modulo value (1 or 2)
REGADR-ACTIVE	Indicates, whether Registration address will be sent 0 = Reg. address will not be sent 1 = Reg. address will be sent

(For REGADR-ACTIVE = 1)

LCNCALLED	Called address field with the first byte containing the length and with the address bytes in ASCII
LCNCALLING	Calling address field with the first byte containing the length and with the address bytes in ASCII
REGIST-ACTIVE	Indicates, whether Registration field will be sent 0 = Reg. field not wanted 2 = Send Reg. field defined by MAKE_REG

S:REGISTC (--)

Send a Registration Request packet on selected channel using values from the following variables:

LCN	Logical group and channel value
SENDQB	Q bit value
SENDDB	D bit value
SENDGFI	GFI modulo value (1 or 2)
REGADR-ACTIVE	Indicates, whether Registration address will be sent 0 = Reg. address will not be sent 1 = Reg. address will be sent

(For REGADR-ACTIVE = 1)

LCNCALLED	Called address field with the first byte containing the length and with the address bytes in ASCII
LCNCALLING	Calling address field with the first byte containing the length and with the address bytes in ASCII
REGIST-ACTIVE	Indicates, whether Registration field will be sent 0 = Reg. field not wanted 2 = Send Reg. field defined by MAKE_REG
SCAUSE	Cause value
SDIAG	Diagnostic value

3.4. MISCELLANEOUS COMMANDS

3.4.1. PROTOCOL TIMERS

The network layer 3 Emulation has timers for Restart, Call, Reset and Clear. These timer values (in tenths of seconds) are stored in the following variables:

T10-VALUE	- RESTART timer
T11-VALUE	- CALL timer
T12-VALUE	- RESET timer
T13-VALUE	- CLEAR timer
T20-VALUE	- RESTART timer
T21-VALUE	- CALL timer
T22-VALUE	- RESET timer
T23-VALUE	- CLEAR timer

3.4.2. REFERENCE EMULATION STATE MACHINES

To ensure correct protocol operation, state machines have been implemented Layer 3. The state machines normally handle the protocol automatically. The user may activate or deactivate the state machines with the following commands:

L3_ON (--)
Activate Layer 3 state machines. There will be automatic responses to received packets.

L3_OFF (--)
Deactivate Layer 3 state machines.

The user also has the ability to set the state machines into any state by using the following commands:

NEW_L3_STATE ({1...4} --)
To change the Layer 3 state machines to the specified state. Valid states are from 1 to 4 inclusively. Invalid states will not be checked and it is the user's responsibility to ensure only the valid states are used.

NEW_LCN_STATE ({1|2|4|5|6|8} --)
To change the Logical Channel state machines to the specified state. Valid states are 1, 2, 4, 5, 6 and 8. Invalid states will not be checked and it is the user's responsibility to ensure only the valid states are used.

3.4.2.1. LAYER 3 STATE MACHINES

The Layer 3 state machines consist of 4 states.

STATE #	STATE	DESCRIPTION
1	IDLE	
2	WAITING OPERATIONAL	Waiting for link operational
3	WAITING RESTART CONFIRM	Waiting for Restart Confirmation
4	LINK UP	Operational state

When the Layer 3 state machines are in State 4 (LINK UP), the logical channel state machines will be accessible. The logical channel state machines are comprised of 6 states.

STATE #	STATE	DESCRIPTION
1	P1 READY	LCN is ready to send or receive call
2	P2 CALLING	LCN has sent Call Request, waiting for Call Connected
4	D1 DATA	Flow control state
5	D2 RESET	LCN has sent Reset Request, waiting for Reset Confirmation
6	P6 CLEARING	LCN has sent Clear Request, waiting for Clear Confirmation
8	D8 REMOTE BUSY	Flow control state with remote busy

4. TEST MANAGER

4.1. EVENT RECOGNITION WORDS

4.1.1. ?RX_PACKET

The **?RX_PACKET** word is for use within the event field to check whether one of a list of Layer 3 events has been received.

Format: `EVENT1 ... EVENTN n ?RX_PACKET`
where n is the total number of events to be checked, and the events can be any one of the following:

EVENT	DESCRIPTION
R*DATAP	Received Data packet
R*RRP	Received RR packet
R*REJP	Received RNR packet
R*CALLREQ	Received Call Request packet
R*CALLCON	Received Call Connected packet
R*CLEARREQ	Received Clear Request packet
R*CLEARCONF	Received Clear Confirmation packet
R*INTREQ	Received Interrupt Request packet
R*INTCONF	Received Interrupt Confirmation packet
R*RESETREQ	Received Reset Request packet
R*RESETCONF	Received Reset Confirmation packet
R*RESTARTREQ	Received Restart Request packet
R*RESTARTCONF	Received Restart Confirmation packet
R*INVPKT	Received Invalid packet
R*DIAGNOSTIC	Received Diagnostic packet

This word is intended to be used in the case where Layer 2 is running automatically and the Layer 3 state machine is not to be executed. This allows the test scenario to have total control of packet responses.

Example:

```
1 STATE{  R*CALLREQ  1  ?RX_PACKET          ( check for Call Request packet )
               ACTION{  ." Call Request received"  CR
               }ACTION
}STATE
```

4.1.2. ?RX

The **?RX** word is used within the event field to check whether one of a list of packets has been received.

Format: `EVENT1 . . . EVENTN n ?RX`
where N is the total number of events, and the event may be any one of the packet events listed in the ?RX_PACKET list.
This word is intended to be used in the case where layer 2 is running automatically and layer 3 State-machine is to be executed too.
That means the testscenario can make use of the L3 Emulator as long as necessary, before getting the control of packet responses.

4.1.3. LCN-MULTIPLEXER

=CN_LCN (lcn-value --)
select a logical channel number for multiplexing. The current connection only listens to packets with this LCN **and** to packets with LCN=0!

?CN_LCN (-- lcn-value)
returns the selected logical channel

=CN_CH (ch-value --)
select a channel structure this connection

?CN_CH (-- ch-value)
returns the selected channel structure

Example:

```
0 STATE_INIT{
      L3_MUX_ON    L3_OFF
      #UNUSED      =CN_LCN
      1            =CN_CH            ( sets the channel structure
                                       to 1. Ie., if the incoming
                                       frame pertains to this
                                       connection, the current
                                       channel structure is updated
                                       to the selected one (CH1))
}STATE_INIT

0 STATE{
      R*RESTARTREQ 1 ?RX_PACKET
      ACTION{
                   MDL_CURRENT =CN_CEI
                   S:RESTARTC
      }ACTION

      R*CALLREQ    1 ?RX_PACKET
      ACTION{
                   M-LCN@ =CN_LCN     (sets the LCN for this
                                       connection with received LCN.
                                       Ie., only packets with the
                                       selected channel number are
                                       routed to this connection)
                   S:CALLC
      }ACTION
}STATE
```

4.2. LIST OF COMMUNICATION VARIABLES

The X.25 program utilizes variables to hold the decoded information from received frames. The following list gives a description of the variables that can be utilitzed the Test Manager action field to analyze the protocol procedure.

PKT-LENGTH	(-- addr)	Length of the received packet.
PKT-POINTER	(-- addr)	Pointer to the first byte of the received packet.
M-GFI	(-- addr)	GFI value (1=Modulo 8, 2=Modulo 128).
M-Q	(-- addr)	Q bit.
M-D	(-- addr)	D bit.
M-LCN	(-- addr)	Pointer of logical group and channel value. (12 bit)
M-REC-PKT-ID	(-- addr)	Packet type identifier byte.
M-RCALLED	(-- addr)	Array of 16 byte which contains called address, first byte is length. The address is stored as ASCII characters.
M-RCALLING	(-- addr)	Array of 16 byte which contains calling address, first byte is length. The address is stored as ASCII characters.
M-RFAC	(-- addr)	Array of 128 byte which contains received facility field. First byte is length.
M-RCUD	(-- addr)	Array of 256 byte which contains received Call or Clear user data field. First byte is length.
M-RCAUSE	(-- addr)	Cause byte.
M-RDIAG	(-- addr)	Diagnostic byte.
M-DIAGNOSTICF	(-- addr)	Pointer of Diagnostic field from diagnostic packet.
M-MORE	(-- addr)	More bit from data packet.
M-PS	(-- addr)	PS value.
M-PR	(-- addr)	PR value.
DATA-LENGTH	(-- addr)	Length of Data field in data packets.
DATA-POINTER	(-- addr)	Pointer to first byte of data field in data packet.

LAYER 3

DATA-LENGTH	(-- addr)	Length of the received data packet field
DATA-POINTER	(-- addr)	Address of the received data packet field
PR	(-- addr)	Received PR sequence
PS	(-- addr)	Received PS sequence
PKT-LENGTH	(-- addr)	Length of received packet layer
PKT-POINTER	(-- addr)	Address of received packer layer
RCALLED	(-- addr)	16 byte received calling address (count prefixed)
RCALLING	(-- addr)	16 byte received called address (count prefixed)
RCAUSE	(-- addr)	Received cause field
RCUD	(-- addr)	128 byte received call user data field
RDIAG	(-- addr)	Received diagnostic field
REC_PKT_ID	(-- addr)	ID byte received call user data field
RECD	(-- addr)	Received D bit
RECM	(-- addr)	Received M bit
RECPKTMOD	(-- addr)	Received packet Modulo ID from GFI
RECQ	(-- addr)	Received Q bit
RFAC	(-- addr)	64 byte received facility field
RIUD	(-- addr)	Received Interrupt User Data byte from Interrupt Request packets
RLCN	(-- addr)	Received LCN value
SCAUSE	(-- addr)	Cause field for transmitted RESET, RESTART, CLEAR packets
SDIAG	(-- addr)	Diagnostic field for transmitted RESET, RESTART, CLEAR packets
SDIAG-EXP	(-- addr)	Diagnostic explanation field. The first byte is the length of the field followed by up to 3 bytes for the explanation field to be sent.
SENDM	(-- addr)	Value of next M bit to be sent
SIUD	(-- addr)	Interrupt User Data buffer used for transmitting Interrupt Request packets
STATE_L3	(-- addr)	Layer 3 state variable
T20_VALUE	(-- addr)	T20 RESTART timer
T21_VALUE	(-- addr)	T21 CALL timer
T22_VALUE	(-- addr)	T22 RESET timer
T23_VALUE	(-- addr)	T23 CLEAR timer

SENDQB (-- addr) SENDQB is a variable which contains the value of the Q bit used for sending packets. The valid value is 0 or 1.

SENDDB (-- addr) SENDDB is a variable which contains the value of the D bit used for sending packets. The valid value is 0 or 1.

SENDGFI (-- addr) SENDGFI is a variable which contains the value of the GFI modulo indicator used for sending packets. The valid value is 0X10 or 0X20.

LAYER 3 - LOGICAL CHANNEL VARIABLES

The logical channel variables access the values pertaining to the current logical channel selection (CH1 to 3).

LCN (-- addr) Selected logical channel number.

LCNSTATE (-- addr) State for Reference emulation.

LCNWINDOW (-- addr) Window size for selected logical channel.

LCNSIZE (-- addr) Size of data field used with DATA command.

NVS (-- addr) LCN Send Sequence variable. This is the next PS that will be sent.

NVR (-- addr) LCN Receive Sequence variable. This is the next PS expected to be received.

NSU (-- addr) LCN sequence number (PS) of the lowest unacknowledged Data Packet that was sent.

LCNCALLING (-- addr) LCN calling address field

LCNCALLED (-- addr) LCN called address field.

COMMANDS

=CN_LCN (lcn-value --)
sets the logical channel number for currently selected connection

?CN_LCN (-- lcn-value)
returns the logical channel number for currently selected connection

=CN_CH (ch-value --)
sets the logical channel for currently selected connection

?CN_CH (-- ch-value)
returns the logical channel for currently selected connection

INDEX

Symbols

?CN_CH, 4-2, 4-5
?CN_LCN, 4-2, 4-5
?RX, 4-1
?RX_PACKET, 4-1
=CALLED, 3-6
=CALLING, 3-7
=CN_CH, 4-2, 4-5
=CN_LCN, 4-2, 4-5
=HUNT_ADDRESS, 3-5
=REDIR_ADDRESS, 3-6

C

CALL, 3-8
CALLS, 3-8
CCITT-80, 3-1
CCITT-84, 3-1
CH1, 3-6
CH2/../CH3, 3-7
CLASS, 3-1
CLASS_NEG, 3-4
CLEAR, 3-8
CLEARS, 3-8
COMMAND, 2-3

D

DATA, 3-9
DATA-LENGTH, 4-3, 4-4
DATA-POINTER, 4-3, 4-4
DEST, 3-6

E

ECHO_CAFAC, 3-2
ECHO_OFF, 3-7
ECHO_ON, 3-7

F

FAST_SEL_OFF, 3-7
FAST_SEL_ON, 3-7
FAST_SEL_RE, 3-4
FAST_SEL_RES, 3-7
FLOW_CONTROL_NEG, 3-4
FREEZE, 2-3

H

HALT, 2-3
HELP, 2-3
HUNT_BUSY, 3-5
HUNT_FREE, 3-5

I

INTERRUPT, 3-9

L

L3_MODULO128, 3-5
L3_MODULO8, 3-5
L3_MODULUS, 3-5
L3_OFF, 3-14
L3_ON, 3-14
LCN, 3-7, 4-5
LCNCALLED, 4-5
LCNCALLING, 4-5
LCNSIZE, 4-5
LCNSTATE, 4-5
LCNWINDOW, 4-5
LOAD_FAC, 3-4

M

M-D, 4-3
M-DIAGNOSTICF, 4-3
M-GFI, 4-3
M-LCN, 4-3
M-MORE, 4-3
M-PR, 4-3
M-PS, 4-3
M-Q, 4-3
M-RCALLED, 4-3
M-RCALLING, 4-3
M-RCAUSE, 4-3

M-RCUD, 4-3
M-RDIAG, 4-3
M-REC-PKT-ID, 4-3
M-RFAC, 4-3
MAKE_CAFAC, 3-2
MAKE_CUD, 3-3
MAKE_FAC, 3-2
MAKE_REG, 3-3
MENU, 2-3
MON_KEYS, 2-3
MONITOR, 2-3

N

NEW_L3_STATE, 3-14
NEW_LCN_STATE, 3-14
NO_CA, 3-2
NO_CAFAC, 3-2
NO_CLASS_NEG, 3-4
NO_FAC, 3-1
NO_FAST_SEL, 3-4
NO_FLOW_NEG, 3-4
NO_HUNT, 3-5
NO_NEG_FAC, 3-4
NO_REDIRECTION, 3-5
NO_REG, 3-3
NO_REGADR, 3-3
NO_REVCH, 3-4
NSU, 4-5
NVR, 4-5
NVS, 4-5

O

ORIG, 3-6

P

PKT-LENGTH, 4-3, 4-4
PKT-POINTER, 4-3, 4-4
PR, 4-4
PS, 4-4

Q

QUIT_COM, 2-3
QUIT_HELP, 2-3
QUIT_MENU, 2-3

R

RCALLED, 4-4
RCALLING, 4-4
RCAUSE, 4-4
RCUD, 4-4
RDIAG, 4-4
REC_PKT_ID, 4-4
RECD, 4-4
RECM, 4-4
RECPKTMOD, 4-4
RECQ, 4-4
REDIR_BUSY, 3-6
REDIR_FREE, 3-6
RESET, 3-8
RESTART, 3-8
RFAC, 4-4
RIUD, 4-4
RLCN, 4-4

S

S:CALLC, 3-11
S:CALLR, 3-10
S:CLEAR, 3-11
S:CLEARC, 3-11
S:DATAP, 3-12
S:DIAG, 3-12
S:INTC, 3-12
S:INTR, 3-12
S:REGISTC, 3-13
S:REGISTR, 3-13
S:REJP, 3-13
S:RESETC, 3-11
S:RESETR, 3-11
S:RESTARTC, 3-10
S:RESTARTR, 3-10
S:RNRP, 3-12

S:RRP, 3-12
SAVE_FAC, 3-4
SCAUSE, 4-4
SDIAG, 4-4
SDIAG-EXP, 4-4
SENDD, 3-9
SENDDB, 4-5
SENDGFI, 4-5
SENDM, 4-4
SENDP, 3-9
SENDQB, 4-5
SET_SIZE, 3-8
SET_WINDOW, 3-7
SIUD, 4-4
STATE_L3, 4-4
SVC, 3-6

T

T10-VALUE, 3-14
T11-VALUE, 3-14
T12-VALUE, 3-14
T13-VALUE, 3-14
T20-VALUE, 3-14
T20_VALUE, 4-4
T21-VALUE, 3-14
T21_VALUE, 4-4
T22-VALUE, 3-14
T22_VALUE, 4-4
T23-VALUE, 3-14
T23_VALUE, 4-4
TM_KEYS, 2-3

U

USER_CAFAC, 3-2
USER_FAC, 3-1
USER_REG, 3-3

Y

YES_CA, 3-2
YES_FAC, 3-1
YES_FAST_SEL, 3-4
YES_HUNT, 3-5
YES_NEG_FAC, 3-4
YES_REDIRECTION, 3-5
YES_REGADR, 3-3
YES_REVCH, 3-4

Inhalt

K1197 Dual Interface Conformance Tests

C73000-M6000-C189

2

3

4

5

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patenterteilung oder GM-Eintragung vorbehalten.

Technische Änderungen vorbehalten

Bestell-Nr. C73000-G6000-C119-1
Bestellort: MWB
Printed in the Federal Republic of Germany
AG 1092 0.02 MWB OI De

ISDN LAYER 3 INTERNATIONAL VARIANT (Q931)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version : V2.0
System Version : MF 2.1

Revision Date : January 1990

1 INTRODUCTION 1-1

2 MESSAGE TYPE IDENTIFIER 2-1

3 INFORMATION ELEMENT TYPE IDENTIFIER 3-1

4 OCTET IDENTIFIER 4-1

5 LAYER 3 TIMER 5-1

6 ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1 Shift IE (I#SHIFT) 6-1
6.2 Congestion Level IE (I#CONG_LEVEL) 6-1
6.3 Repeat Indicator IE (I#REP_IND) 6-2
6.4 Segmented Message IE (I#SEG_MSG) 6-2
6.5 Bearer Capability IE (I#BEARER_CAP) 6-3
6.6 Cause IE (I#CAUSE) 6-7
6.7 Call Identity IE (I#CALL_ID) 6-9
6.8 Call State IE (I#CALL_STATE) 6-10
6.9 Channel Identification IE (I#CHANNEL_ID) 6-11
6.10 CCITT-Standardized Facility IE (I#FACIL) 6-12
6.11 Progress Indicator IE (I#PROGREES_ID) 6-15
6.12 Network-Specific Facility IE (I#NETWORK_FACIL) 6-16
6.13 Notification Indicator IE (I#NOTIFIC_IND) 6-16
6.14 Display IE (I#DISPLAY) 6-17
6.15 Date/Time IE (I#DATE) 6-17
6.16 Keypad IE (I#KEYPAD) 6-17
6.17 Signal IE (I#SIGNAL) 6-18
6.18 Switchhook IE (I#SWITCHHOOK) 6-18
6.19 Feature Activation IE (I#FEATURE_ACT) 6-19
6.20 Feature Indication IE (I#FEATURE_IND) 6-19
6.21 Information Rate IE (I#INFO_RATE) 6-19
6.22 End-to-End Transit Delay IE (I#END_DELAY) 6-21
6.23 Transit Delay Selection and Indication IE (I#TRANS_DEL) 6-21

6.24 Packet Layer Binary Parameters IE (I#PACKET_BIN) 6-22
6.25 Packet Layer Window Size IE (I#PACKET_WIN) 6-22
6.26 Packet Size IE (I#PACKET_SIZE) 6-23
6.27 Calling Number IE (I#CALLING_NUM) 6-23
6.28 Calling Subaddress IE (I#CALLING_SAD) 6-24
6.29 Called Number IE (I#CALLED_NUM) 6-24
6.30 Called Subaddress IE (I#CALLED_SAD) 6-25
6.31 Redirecting Number IE (I#REDIRECTING_NUM) 6-25
6.32 Transit Network Selection IE (I#TRANS_NW_SEL) 6-26
6.33 Restart Indicator IE (I#RESTART_IND) 6-27
6.34 Low Layer Compatability IE (I#LOW_LAY_COMP) 6-28
6.35 High Layer Compatability IE (I#HI_LAY_COMP) 6-33
6.36 User User Information IE (I#UU_INFO) 6-34
6.37 Escape for Extension IE (I#ESCAPE) 6-34

1 INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC / *COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2 MESSAGE TYPE IDENTIFIER

Message Identifier	CCITT Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#CON_CON	Congestion Control
M#FAC	Facility
M#INFO	Info
M#NOTIFY	Notify
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry
M#SEGMENT	Segmented Message

3 INFORMATION ELEMENT TYPE IDENTIFIER

IE Identifier	CCITT Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#REP_IND	Repeat Indicator
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#SWITCHHOOK	Switchhook
I#FEATURE_ACT	Feature Activation
I#FEATURE_IND	Feature Indication
I#INFO_RATE	Information Rate
I#END_DELAY	End-to-End Transit Delay
I#TRANS_DEL	Transit Delay Selection and Indication
I#PACKET_BIN	Packet Binary Parameters
I#PACKET_WIN	Packet Binary Window Size
I#PACKET_SIZE	Packet Size
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#TRANS_NW_SEL	Transit Network Selection
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information
I#ESCAPE	Escape for Extension

4 OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_3.1, OCTET_3.2, OCTET_3.3,
OCTET_4, OCTET_4A, OCTET_4B,
OCTET_5, OCTET_5A, OCTET_5B, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D,
OCTET_6, OCTET_6A, OCTET_6.1, OCTET_6.2, OCTET_6.3,
OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8,
OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5,
OCTET_7, OCTET_7A,
OCTET_8

5 LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T301	#T301	180	180
Timer T302	#T302	15	15
Timer T303	#T303	4	4
Timer T304	#T304	15	20
Timer T305	#T305	30	30
Timer T306	#T306	30	30
Timer T307	#T307	180	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	10	10
Timer T312	#T312	6	6
Timer T313	#T313	4	4
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	60	60
Timer T318	#T318	4	4
Timer T319	#T319	4	4
Timer T320	#T320	30	30
Timer T321	#T321	30	30
Timer T322	#T322	4	4
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6 ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1 Shift IE (I#SHIFT)

Selector Name and related Constants	CCITT Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	Codeset Specified in Shift IE
#CODESET0	*I.451 (Q.931) IE*
#CODESET5	*National Use IE*
#CODESET6	*Local Network Specific IE*
#CODESET7	*User Specific IE*

6.2 Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	CCITT Reference
->CL_CONGESTION_LEVEL	Congestion Level
#RECEIVER_READY	*Receiver Ready*
#RECEIVER_NOT_READY	*Receiver not Ready*

6.3 Repeat Indicator IE (I#REP_IND)

Selector Name and related Constants	CCITT Reference
->RP_INDICATION #PRIORITIZED	Repeat Indication *Prioritized List*

6.4 Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.5 Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_7

Selector Name and related Constants	CCITT Reference
->BC_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for present network*
->BC_TRANSFER_CAP	Information Transfer Capability, Octet 3
#SPEECH	*Speech*
#UNRESTRICTED	*Unrestricted Digital Information*
#RESTRICTED	*Restricted Digital Information*
#3.1KHZ_AUDIO	*3.1 kHz Audio*
#7KHZ_AUDIO	*7 kHz Audio*
#VIDEO	*Video*
->BC_TRANSFER_MODE	Transfer Mode, Octet 4
#CIRCUIT_MODE	*Circuit Mode*
#PACKET_MODE	*Packet Mode*
->BC_TRANSFER_RATE	Information Transfer Rate, Octet 4
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_STRUCTURE	Structure, Octet 4a
#DEFAULT	*Default*
#8KHZ_INTEGRITY	*8 kHz Integrity*
#SDU_INTEGRITY	*Service Data Unit Integrity*
#UNCONSTRUCTURED	*Unstructured*
->BC_CONFIGURATION	Configuration, Octet 4a
#POINT_TO_POINT	*Point-to-point*
->BC_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->BC_SYMMETRY	Symmetry, Octet 4a
#BIDIRECT_SYMMENTRIC	*Bidirectional Symmetric*

Selector Name and related Constants	CCITT Reference
->BC_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->BC_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.7XX 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->BC_SYNC/ASYNC	Synchronous or Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->BC_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->BC_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6 and X.1*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*

Selector Name and related Constants	CCITT Reference
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation V.6 and X.1*
#1.2/0.075KBIT/S	*1.2/0.075 kbit/s Recommendation V.6 and X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.75KBIT/S	*0.75 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->BC_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*not used*
#INT_8KBIT/S	*8 kbit/s*
#INT_16KBIT/S	*16 kbit/s*
#INT_32KBIT/S	*32 kbit/s*
->BC_NIC_ON_TX	Network internal clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->BC_NIC_ON_RX	Network internal clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->BC_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*No Flow Control required*
#REQUIRED	*Flow Control required*
->BC_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept Flow Control*
#ACCEPT	*Can accept Flow Control*
->BC_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->BC_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->BC_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*
->BC_LLI_NEG	Logical Link negotiation, Octet 5b.2
#DEFAULT	*Default, LLI=256 only*
#FULL	*Full protocol negotiation*

Selector Name and related Constants	CCITT Reference
->BC_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assigner'* *Originator is 'Assigner only'*
->BC_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out-band negotiation, Octet 5b.2 *Negotiation is done 'Out-band'* *Negotiation is done 'In-band'*
->BC_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of stop bits, Octet 5c *Not used* *1 bit* *1.5 bits* *2 bits*
->BC_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of data bits, Octet 5c *Not used* *5 bits* *7 bits* *8 bits*
->BC_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity information, Octet 5c *odd* *even* *none* *forced to 0* *forced to 1*
->BC_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
->BC_MODEM_TYPE	Modem type, Octet 5d
->BC_LAYER2_ID #LAYER2	Layer Identification, Octet 6 *User Info Layer 2 Protocol*
->BC_L2_PROTOCOL #Q.921 #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation Q.921/I.441* *Recommendation X.25, Link Layer*
->BC_LAYER3_ID #LAYER3	Layer 3 Identification, Octet 7 *User Info Layer 3 Protocol*
->BC_L3_PROTOCOL #Q.931 #X.25_PACKET	Layer 3 Protocol Identifier, Octet 7 *Recommendation Q.921/I.441* *Recommendation X.25, Packet Layer*

Note 1 : Octet 5b.1 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V120

6.6 Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_5

Selector Name and related Constants	CCITT Reference
->C_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other international Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->C_LOCATION	Location, Octet 3
#USER	*User*
#LOACAL_PRIVATE	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#TRANSIT	*Transit Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->C_RECOMMENDATION	Recommendation, Octet 3a
#REC_Q.931	*Recommendation Q.931*
#REC_X.21	*Recommendation X.21*
#REC_X.25	*Recommendation X.25*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL0	*Normal Event Class 0*
->C_CAUSE_VALUE	Cause Value, Octet 4
#UNASSIGNED_NUMBER	*Unassigned Number*
#NO_ROUTE_TO_TRANSIT	*No Route to Transit Network*
#NO_ROUTE_TO_DEST	*No Route to Destination*
#CHANNEL_UNACCEPTABLE	*Channel Unacceptable*
#CALL_AWARDED	*Call awarded*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NON_SELECTED_CLEAR	*Non selected user clearing*

Selector Name and related Constants	CCITT Reference
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility Rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
->C_CAUSE_CLASS	Cause Class, Octet 4
#RESOURCE_UNAVAIL	*Resource not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NO_CHANNEL_AVAIL	*No Circuit/Channel Available*
#OUT_OF_ORDER	*Network Out of Order*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access Information Discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#QUALITY_UNAVAIL	*Quality of service unavailable*
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer Capability not Authorized*
#BEARER_UNAVAIL	*Bearer Capability not Available*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#CHANNEL_TYPE_UNIMPL	*Channel Type not Implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#ONLY_RESTRICTED	*Restricted only*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INVALID_MESSAGE	*Invalid Message*
->C_CAUSE_VALUE	Cause Value, Octet 4
#INVALID_CALL_REF	*Invalid Call Reference Value*
#CHANNEL_NONEXISTANT	*Channel not Existant*
#CALL_ID_NONEXISTANT	*Call Identity does not Exist*
#CALL_ID_IN_USE	*Call Identity in Use*
#NO_CALL_SUSPENDED	*No call suspended*
#CALL_CLEARED	*Call has been cleared*
#INCOMPATIBLE_DEST	*Incompatible Destination*
#INV_TRANS_NET	*Invalid transit network selection*
#INVALID_MESSAGE_UNSPEC	*Invalid Message*

Selector Name and related Constants	CCITT Reference
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on Timer Expiry* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.7 Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.8 Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->CS_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*
#CALL_ABORT	*Call Abort*
#OVERLAP_RECEIVING	*Overlap Receiving*
#RESTART_REQUEST	*Restart request*
#RESTART	*Restart*

6.9 Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_3.3

Selector Name and related Constants	CCITT Reference
->CID_INT_PRESENT #EXPLICIT #IMPLICIT	Interface Identifier Present, Octet 3 *Explicitly Identified* *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Other Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #D_CHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *D-Channel Indentified* *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_INT_ID	Interface identifier, Octet 3.1 *
->CID_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3.2 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 3.2 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS #H0_CHANNEL_UNITS #H11_CHANNEL_UNITS #H12_CHANNEL_UNITS	Channel or Map Type, Octet 3.2 *B-Channel Units* *H0-Channel Units* *H11-Channel Units* *H12-Channel Units*
->CID_MAP	Slot Map, Octet 3.3
->CID_NUMBER	Channel number, Octet 3.3

6.10 CCITT-Standardized Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	CCITT Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	CCITT Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #NO_SUPPORT	Error Value, Octet 6.5 (Note 3) *Not supported*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	CCITT Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.11 Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PI_CODING_STANDARD	Coding Standard, Octet 3.2
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->PI_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVAT	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->PI_DESCRIPTION	Progress Description, Octet 4
#NOT_END_TO_END	*Call is not end-to-end ISDN*
#DEST_NON_ISDN	*Destination Address is non-ISDN*
#ORIG_NON_ISDN	*Origination Address is non-ISDN*
#RETURNED_TO_ISDN	*Call has Returned to the ISDN*
#INBAND_INFO_AVAIL	*In-band Info now Available*

6.12 Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_4

Selector Name and related Constants	CCITT Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3.1 *User specified* *National network id.* *International network id.*
->NF_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3.1 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->NF_NET_ID	Network Identification, Octet 3.2 *
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.13 Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change*

6.14 Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->D_DISPLAY	Display, Octet 3 *

6.15 Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8

Selector Name and related Constants	CCITT Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7
->DT_SECOND	Second, Octet 8

6.16 Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->K_KEYPAD	Keypad, Octet 3 *

6.17 Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*

6.18 Switchhook IE (I#SWITCHHOOK)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->SW_VALUE	Switchhook Value, Octet 3
#ON_HOOK	*On-hook*
#OFF_HOOK	*Off-hook*

6.19 Feature Activation IE (I#FEATURE_ACT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A

Selector Name and related Constants	CCITT Reference
->FA_ID_NUMBER	Feature identifier number, Octet 3
->FA_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a

6.20 Feature Indication IE (I#FEATURE_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	CCITT Reference
->FI_ID_NUMBER	Feature identifier number, Octet 3
->FI_ID_NUMBER_CONT	Feature id. number continuation, Octet 3a
->FI_STATUS_IND #DEACTIVATED #ACTIVATED #PROMPT #PENDING	Status indicator, Octet 4 *Feature is inactive* *Feature is active* *Feature prompt* *Feature is pending*

6.21 Information Rate IE (I#INFO_RATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6

Selector Name and related Constants	CCITT Reference
->IR_INCOMING #75_B/S #150_B/S #300_B/S #600_B/S #1200_B/S #2400_B/S #4800_B/S #9600_B/S #19200_B/S #48000_B/S	Incoming information rate, Octet 3 *75 bit/s* *150 bit/s* *300 bit/s* *600 bit/s* *1200 bit/s* *2400 bit/s* *4800 bit/s* *9600 bit/s* *19200 bit/s* *48000 bit/s*

Selector Name and related Constants	CCITT Reference
->IR_INCOMING	Incoming information rate, Octet 3
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_OUTGOING	Outgoing information rate, Octet 4
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_MIN_INCOMING	Minimum incoming info rate, Octet 5
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*
->IR_MIN_INCOMING	Minimum incoming info rate, Octet 5
#75_B/S	*75 bit/s*
#150_B/S	*150 bit/s*
#300_B/S	*300 bit/s*
#600_B/S	*600 bit/s*
#1200_B/S	*1200 bit/s*
#2400_B/S	*2400 bit/s*
#4800_B/S	*4800 bit/s*
#9600_B/S	*9600 bit/s*
#19200_B/S	*19200 bit/s*
#48000_B/S	*48000 bit/s*

6.22 End-to-End Transit Delay IE (I#END_DELAY)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B

Selector Name and related Constants	CCITT Reference
->ED_CUM_DELAY	Cumultative transit delay value, Octet 3, 3a, 3b
->ED_CUM_DELAY_3	Cumultative trans. del. val., bit 16–15 (Octet 3)
->ED_CUM_DELAY_3A	Cumultative trans. del. val., bit 14–8 (Octet 3a)
->ED_CUM_DELAY_3B	Cumultative trans. del. val., bit 7–1 (Octet 3b)
->ED_REQ_DELAY	Requested transit delay value, Octet 4, 4a, 4b
->ED_REQ_DELAY_4	Requested trans. del. val., bit 16–15 (Octet 4)
->ED_REQ_DELAY_4A	Requested trans. del. val., bit 14–8 (Octet 4a)
->ED_REQ_DELAY_4B	Requested trans. del. val., bit 7–1 (Octet 4b)
->ED_MAX_DELAY	Maximum transit delay value, Octet 5, 5a, 5b
->ED_MAX_DELAY_5	Maximum trans. del. val., bit 16–15 (Octet 5)
->ED_MAX_DELAY_5A	Maximum trans. del. val., bit 14–8 (Octet 5a)
->ED_MAX_DELAY_5B	Maximum trans. del. val., bit 7–1 (Octet 5b)

6.23 Transit Delay Selection and Indication IE (I#TRANS_DEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B

Selector Name and related Constants	CCITT Reference
->TD_SEL/IND	Transit delay sel/ind value, Octet 3, 3a, 3b
->TD_SEL/IND_3	Transit delay sel/ind val., bit 16–15 (Octet 3)
->TD_SEL/IND_3A	Transit delay sel/ind val., bit 14–8 (Octet 3a)
->TD_SEL/IND_3B	Transit delay sel/ind val., bit 7–1 (Octet 3b)

6.24 Packet Layer Binary Parameters IE (I#PACKET_BIN)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->PB_FAST_SELECT #NOT_REQUESTED_0 #NOT_REQUESTED_1 #REQUESTED_NOT_REST #REQUESTED_REST	Fast select, Octet 3 *Not requested* *Not requested* *Requested, no restriction* *Requested, restriction*
->PB_EXP_DATA #NO_REQUEST #REQUEST_IND	Expedited data, Octet 3 *No request / request denied* *Request indicated / accepted*
->PB_DELIVERY_CONF #LINK_BY_LINK #END_TO_END	Delivery confirmation, Octet 3 *Link-by-link confirmation* *End-to-end confirmation*
->PB_MODULUS #MODULUS_8 #MODULUS_128	Modulus, Octet 3 *Modulus 8 sequencing* *Modulus 128 sequencing*

6.25 Packet Layer Window Size IE (I#PACKET_WIN)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PWS_FORWARD	Forward value, Octet 3
->PWS_BACKWARD	Backward value, Octet 4

6.26 Packet Size IE (I#PACKET_SIZE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->PS_FORWARD	Forward value, Octet 3
->PS_BACKWARD	Backward value, Octet 4

6.27 Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLGN_NUMBER_TYPE	Type of Number, Octet 3
#UNKNOWN	*Unknown*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
#SUBSCRIBER	*Subscriber Number*
#ABBREVIATED	*Abbreviated Number*
->CLGN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#UNKNOWN_PLAN	*Unknown*
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#DATA_PLAN	*Data Numbering Plan X.121*
#TELEX_PLAN	*Telex Numbering Plan F.69*
#NATIONAL_PLAN	*National Numbering Plan*
#PRIVATE_PLAN	*Private Numbering Plan*
->CLGN_PRESENTATION	Presentation Indicator, Octet 3
#PRESENT_ALLOWED	*Presentation Allowed*
#PRESENT_RESTRICTED	*Presentation Restricted*
#NUMBER_UNAVAIL	*Number Not Available Due to Interworking*
->CLGN_SCREENING	Screening indicator, Octet 3a
#UNSCREENED	*User-Provided, Not Screened*
#VERIFY_PASSED	*User-Provided, Verified and Passed*
#VERIFY_FAILED	*User-Provided, Verified and Failed*
#NETWORK_PROVIDED	*Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.28 Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.29 Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLDN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber Number* *Abbreviated Number*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.30 Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->CLDS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.31 Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	CCITT Reference
->RDGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber number* *Abbreviated Number*
->RDGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*

Selector Name and related Constants	CCITT Reference
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->RDGN_REASON #CALL_FORW_BUSY #CALL_FORW_NO_REPLY #CALLED_OUT_OF_ORDER #FORW_BY_CALLED #UNCOND_OR_SYS	Reason for redirection, Octet 3b *Call forwarding or called DTE busy* *Call forwarding no reply* *Called DTE out of order* *Forwarding by called DTE* *Unconditional or systematic red.*
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.32 Transit Network Selection IE (I#TRANS_NW_SEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->TR_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3 *User specified* *National network id.* *International network id.*
->TR_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->TR_NET_ID	Network Identification, Octet 4 *

6.33 Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	CCITT Reference
->RI_CLASS	Class of Channel or Interface, Octet 3
#INDICATED_CHANNEL	*Indicated Channel*
#SINGLE_INTERFACE	*Single Interface*
#ALL_INTERFACES	*All Interface*

6.34 Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	CCITT Reference
->LL_CODING_STANDARD	Coding Standard, Octet 2.1 or 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->LL_TRANSFER_CAP	Information Transfer Capability, Octet 3
#SPEECH	*Speech*
#UNRESTRICTED	*Unrestricted Digital Information*
#RESTRICTED	*Restricted Digital Information*
#3.1KHZ_AUDIO	*3.1 kHz Audio*
#7KHZ_AUDIO	*7 kHz Audio*
#VIDEO	*Video*
->LL_NEGOTIATION_IND	Negotiation indicator, Octet 3a
#NEG_NOT_POSSIBLE	*Out-band negotiation not possible*
#NEG_POSSIBLE	*Out-band negotiation possible*
->LL_TRANSFER_MODE	Transfer Mode, Octet 4
#CIRCUIT_MODE	*Circuit Mode*
#PACKET_MODE	*Packet Mode*
->LL_TRANSFER_RATE	Information Transfer Rate, Octet 4
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->LL_STRUCTURE	Structure, Octet 4a
#DEFAULT	*Default*
#8KHZ_INTEGRITY	*8 kHz Integrity*
#SDU_INTEGRITY	*Service Data Unit Integrity*
#UNSTRUCTURED	*Unstructured*
->LL_CONFIGURATION	Configuration, Octet 4a
#POINT_TO_POINT	*Point-to-point*

Selector Name and related Constants	CCITT Reference
->LL_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->LL_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->LL_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#NO_ADD_L1_PROTOCOL	*No Additional Layer 1 Protocol*
#RATE_ADAPTION	*Rate Adaption*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.711 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*

Selector Name and related Constants	CCITT Reference
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.75KBIT/S	*0.75 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#INT_NOT_USED	*Not Used*
#INT_8KBIT/S	*8kbit/s*
#INT_16KBIT/S	*16kbit/s*
#INT_32KBIT/S	*32kbit/s*
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*Flow Control not required*
#REQUIRED	*Flow Control Required*
->LL_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot Accept Flow Control*
#ACCEPT	*Can Accept Flow Control*
->LL_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->LL_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->LL_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*

Selector Name and related Constants	CCITT Reference
->LL_LLI_NEG #DEFAULT #FULL	Logical Link negotiation, Octet 5b.2 *Default, LLI=256 only* *Full protocol negotiation*
->LL_ASSIGNER #DEFAULT #ONLY_ASSIGNER	Assignor/Assignee, Octet 5b.2 *Originator is 'Default assigner'* *Originator is 'Assigner only'*
->LL_IN/OUT_BAND #OUT_BAND #IN_BAND	In/Out-band negotiation, Octet 5b.2 *Negotiation is done 'Out-band'* *Negotiation is done 'In-band'*
->LL_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of Stop Bits, Octet 5c *Not used* *1 Bit* *1.5 Bits* *2 Bits*
->LL_DATA_BITS #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of Data Bits, Octet 5c *5 Bits* *7 Bits* *8 Bits*
->LL_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity Information, Octet 5c *Odd* *Even* *None* *Forced to 0* *Forced to 1*
->LL_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
->LL_MODEM_TYPE	Modem type, Octet 5d
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #ISO_1745 #Q.921 #X.25_LINK #X.25_MULTILINK #EXTENDED_LAPB #HDLC_ARM #HDLC_NRM #HDLC_ABM #LAN_LLC #X.75	Layer 2 Protocol Identifier, Octet 6 *Basic mode ISO 1745* *Recommendation Q.921 (i.441)* *Recommendation X.25, Link level* *Recommendation X.25 Multilink* *Extended LAPB; half duplex (T.71)* *HDLC ARM (ISO 4335)* *HDLC NRM (ISO 4335)* *HDLC ABM (ISO 4335)* *LAN Logical link control (ISO 8802/2)* *Rec. X.75, single link procedure*

Selector Name and related Constants	CCITT Reference
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET #ISO_8208 #ISO_8348 #ISO_8473 #T.70	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer* *ISO 8208* *ISO 8348* *ISO 8473* *Rec. T.70, minimum network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.35 High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A

Selector Name and related Constants	CCITT Reference
->HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->HL_INTERPRETATION	Interpretation, Octet 3
#FIRST_CHARACTER	*First HL Characteristics to be Used*
->HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
->HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#MAINTENANCE	*Reserved for Maintenance (Note 1)*
#MANAGEMENT	*Reserved for Management (Note 1)*
->HL_CHARACTER_EXT	Extended High Layer Characteristics, Octet 4a
#TELEPHONY	*Telephony*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application*
#NOT_AVAIL1	*Not available for assignment*
#NOT_AVAIL2	*Not available for assignment*

Note 1 : Only in this case, Octet 4a may follow

6.36 User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->UU_PROTOCOL_DISCR	Protocol discriminator, Octet 3
#USER_SPECIFIC_PD	*User specific protocol*
#OSI_HIGHER_LAYER	*OSI Higher Layer Protocols*
#X.244	*X.244*
#SYS_MNG_CONVERG	*System mng. convergence function*
#IA5_CHAR	*IA5 characters*
#V.120	*reserved for V.120 rate adaption*
#Q.931_CALL_CONTROL	*Q931 Call Control Messages*
->UU_USER_INFO	User-User Information, Octet 4 *

6.37 Escape for Extension IE (I#ESCAPE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	CCITT Reference
->EX_IE_ID	Information element identifier, Octet 3
->EX_ESCAPE	Contents of information element, Octet 4 *

ISDN LAYER 3 NATIONAL VARIANT (AT&T : 5ESS)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version : V2.0
System Version : MF 2.1

Revision Date : August, 1990

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

2.1. ONE-OCTET MESSAGES 2-1

2.2. TWO-OCTET MESSAGES 2-2

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

3.1. CODESET 0 IES 3-1

3.2. CODESET 6 IES 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. CODESET 0 IES 6-1
6.1.1. Shift IE (I#SHIFT) 6-1
6.1.2. Bearer Capability IE (I#BEARER_CAP) 6-2
6.1.3. Cause IE (I#CAUSE) 6-3
6.1.4. Call State IE (I#CALL_STATE) 6-5
6.1.5. Channel Identification IE (I#CHANNEL_ID) 6-5
6.1.6. Progress Indicator IE (I#PROGRESS_IND) 6-6
6.1.7. Terminal Capabilities IE (I#TERM_CAPAB) 6-6
6.1.8. Keypad IE (I#KEYPAD) 6-6
6.1.9. Signal IE (I#SIGNAL) 6-7
6.1.10. Switchhook IE (I#SWITCHHOOK) 6-7
6.1.11. Calling Number IE (I#CALLING_NUM) 6-8
6.1.12. Called Number IE (I#CALLED_NUM) 6-8
6.1.13. Restart Indicator IE (I#RESTART_IND) 6-9
6.1.14. Low Layer Compatability IE (I#LOW_LAY_COMP) 6-10

6.2. CODESET 6 IES 6-12
6.2.1. User Code IE (I#USER_CODE) 6-12
6.2.2. Associated Type IE (I#ASSOC_TYPE) 6-12
6.2.3. Selected Call Appearance IE (I#SELECT_CALL) 6-12
6.2.4. Origination Call Appearance IE (I#ORIG_CALL) 6-13
6.2.5. Destination Call Appearance IE (I#DEST_CALL) 6-13
6.2.6. Endpoint Identifier IE (I#ENDPOINT_ID) 6-13
6.2.7. Keypad Control IE (I#KEYPAD_CTRL) 6-13
6.2.8. Other Call Reference IE (I#OTHER_CR) 6-14
6.2.9. Feature Activation IE (I#FEATURE_ACT) 6-14

6.2.10. Feature Indication IE (I#FEATURE_IND) 6-15
6.2.10.1. Further Octets for Status Type: #FEATURE_NUMBER_STATUS 6-15
6.2.10.2. Further Octets for Status Type: #FEATURE_BUTTON_STATUS 6-16
6.2.10.3. Further Octets for Status Type: #MULTIPLE_BUTTON_STATUS 6-16
6.2.10.4. Further Octets for Status Type: #MAINTENANCE_STATUS ... 6-16
6.2.10.5. Further Octets for Status Type: #MAINTENANCE_STATUS ... 6-16
6.2.11. Adjunct Control IE (I#ADJUNCT_CTRL) 6-17
6.2.12. Display Control IE (I#DISPLAY_CTRL) 6-17
6.2.13. Display Field IE (I#DISPLAY_FLD) 6-18
6.2.14. Management IE (I#MANAGEMENT) 6-19
6.2.14.1. Parameter Field for Parameter Identifier: #SERVICE_PROFILE_ID 6-20
6.2.14.2. Parameter Field for Parameter Identifier: #ENDPOINT_ID 6-20
6.2.14.3. Parameter Field for Parameter Identifier: #CALL_REF_ID 6-20
6.2.14.4. Parameter Field for Parameter Identifier: #CALL_STATE_ID .. 6-21
6.2.14.5. Parameter Field for Parameter Identifier: #CHANNEL_ID 6-21
6.2.14.6. Parameter Field for Parameter Identifier: #SERVICE_STATE .. 6-21
6.2.14.7. Parameter Field for Parameter Identifier: #SERVICE_MSG_ERROR 6-22
6.2.14.8. Parameter Field for Parameter Identifier: #LOOPBACK_LOCATION 6-22
6.2.14.9. Parameter Field for Parameter Identifier: #SUPPL_CAPABILITIES 6-22
6.2.14.10. Parameter Field for Parameter Identifier: #CAUSE_ID 6-23
6.2.14.11. Parameter Field for Other Parameter Identifier 6-23

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

2.1. ONE-OCTET MESSAGES

Message Identifier	AT&T Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#INFO	Info
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

2.2. TWO-OCTET MESSAGES

Message Identifier	AT&T Reference
M#CONF	Conference
M#CONF_ACK	Conferene Acknowledge
M#CONF_REJ	Conference Reject
M#TRANS	Transfer
M#TRANS_ACK	Transfer Acknowledge
M#TRANS_REJ	Transfer Reject
M#ASSOC	Associated
M#ASSOC_ACK	Associated Acknowledge
M#DROP	Drop
M#DROP_ACK	Drop Acknowledge
M#DROP_REJ	Drop Reject
M#HOLD	Hold
M#HOLD_ACK	Hold Acknowledge
M#HOLD_REJ	Hold Reject
M#RECONN	Reconnect
M#RECONN_ACK	Reconnect Acknowledge
M#RECONN_REJ	Reconnect Reject
M#REDIR	Redirect
M#MAN_INFO	Management Information

3. INFORMATION ELEMENT TYPE IDENTIFIER

3.1. CODESET 0 IES

IE Identifier	AT&T Reference
I#SHIFT	Codeset Shift
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#PROGRESS_IND	Progress Indicator
I#TERM_CAPAB	Terminal Capabilities
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#SWITCHHOOK	Switchhook
I#CALLING_NUM	Calling Party Number
I#CALLED_NUM	Called Party Number
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability

3.2. CODESET 6 IES

IE Identifier	AT&T Reference
I#USER_CODE	User Code
I#ASSOC_TYPE	Associated Type
I#SELECT_CALL	Selected Call Appearance
I#ORIG_CALL	Origination Call Appearance
I#DEST_CALL	Destination Appearance
I#ENDPOINT_ID	Endpoint Identidier
I#KEYPAD_CTRL	Keypad Control
I#OTHER_CR	Other Call Reference
I#FEATURE_ACT	Feature Activation
I#FEATURE_IND	Feature Indication
I#ADJUNCT_CTRL	Adjunct Control
I#DISPLAY_CTRL	Display Control
I#DISPLAY_FLD	Display Field
I#MANAGEMENT	Management

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,			
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,			
OCTET_6,				
OCTET_7,	OCTET_7C,	OCTET_7D,	OCTET_7E,	OCTET_7F,
OCTET_7G,	OCTET_7H,	OCTET_7I,	OCTET_7I.1,	OCTET_7I.2,

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)
Timer T302	#T302	10
Timer T303	#T303	4
Timer T305	#T305	4
Timer T308	#T308	4
Timer T310	#T310	4
General Purpose Timer 1	#USER1	10
General Purpose Timer 2	#USER2	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. CODESET 0 IES

6.1.1. Shift IE (I#SHIFT)

Selector Name and related Constants	AT&T Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	Codeset Specified in Shift IE
#CODESET0	*Q.931 IE*
#CODESET5	*National Use IE*
#CODESET6	*Local Network Specific IE*
#CODESET7	*User Specific IE*

1.1.2. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_5A, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->BC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate, Octet 4 *Used for packet mode* *64 kbit/s*
->BC_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->BC_L1_PROTOCOL #RATE_ADAPTION_V110/X30 #G.711_ULAW	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption V.110/X.30* *Recommendation G.711 u-law*
->BC_SYNC/ASYNC #SYNCHRONOUS	Synchronous or Asynchronous, Octet 5a *Synchronous*
->BC_USER_RATE #56KBIT/S	User rate, Octet 5a *56kbit/s* *Recommendation V.6*
->BC_LAYER2_ID #LAYER2	Layer Identification, Octet 6 *User Info Layer 2 Protocol*
->BC_L2_PROTOCOL #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation X.25, Link Layer*
->BC_LAYER3_ID #LAYER3	Layer 3 Identification, Octet 7 *User Info Layer 3 Protocol*
->BC_L3_PROTOCOL #X.25_PACKET	Layer 3 Protocol Identifier, Octet 7 *Recommendation X.25, Packet Layer*

6.1.3. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4,

Selector Name and related Constants	AT&T Reference
->C_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->C_LOCATION #USER #LOACAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Transit Network* *Remote Public Network* *Remote Private Network* *International Network* *Network Beyond Interworking Point*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER #NO_ROUTE_TO_TRANSIT	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned Number* *No Route to Transit Network*
->C_CAUSE_CLASS #NORMAL1 ->C_CAUSE_VALUE #NORMAL_CLEARING #USER_BUSY #NO_USER_RESPOND #CALL_REJECTED #NUMBER_CHANGED #INVALID_NUMBER_FORMAT #FACILITY_REJECTED #STATUS_ENQ_RESPONSE #NORMAL_UNSPECIFIED	Cause Class, Octet 4 *Normal Event Class 1* Cause Value, Octet 4 *Normal call clearing* *User Busy* *No User Responding* *Call Rejected* *Number Changed* *Invalid Number Format* *Facility Rejected* *Response to STATUS ENQUIRY* *Normal, Unspecified*

Selector Name and related Constants	AT&T Reference
->C_CAUSE_CLASS #RESOURCE_UNAVAIL ->C_CAUSE_VALUE #NO_CHANNEL_AVAIL #QUEUED #TEMPORARY_FAILURE #SWITCH_CONGESTION #ACCESS_INFO_DISCARD	Cause Class, Octet 4 *Resource not Available* Cause Value, Octet 4 *No Circuit/Channel Available* *Queued* *Temporary Failure* *Switching Equipment Congestion* *Access Information Discarded*
->C_CAUSE_CLASS #SERVICE_UNAVAIL ->C_CAUSE_VALUE #FAC_NOT_SUBSCRIBED #OUTGOING_BARRED #INCOMING_BARRED #BEARER_UNAVAIL #SERVICE_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Service not Available* Cause Value, Octet 4 *Requested facility not subscribed* *Outgoing calls barred* *Incoming calls barred* *Bearer Capability not Available* *Service not Available*
->C_CAUSE_CLASS #SERVICE_UNIMPL ->C_CAUSE_VALUE #BEARER_SERVICE_UNIMPL #CHANNEL_TYPE_UNIMPL #FAC_UNIMPL	Cause Class, Octet 4 *Service not implemented* Cause Value, Octet 4 *Bearer capability not implemented* *Channel Type not Implemented* *Requested facility not implemented*
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTANT #INCOMPATIBLE_DEST #INV_TRANS_NET	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Incompatible Destination* *Invalid transit network selection*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #INVALID_IE_CONTENTS #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *Invalid IE Contents* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Interworking*

6.1.4. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#RELEASE_REQUEST	*Release Request*

6.1.5. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->CID_INT_PRESENT	Interface Identifier Present, Octet 3
#IMPLICIT	*Implicitly Identified*
->CID_INT_TYPE	Interface Type, Octet 3
#BASIC_INTERFACE	*Basic Interface*
->CID_PREF/EXCL	Preferred/Exclusive, Octet 3
#EXCLUSIVE	*Exclusive*
#PREFERRED	*Preferred*
->CID_DCHANNEL	D-channel Indicator, Octet 3
#D_CHANNEL	*D-Channel Indentified*
#NOT_D_CHANNEL	*Not D-Channel*
->CID_INFO_CHAN_SEL	Information Channel Selection, Octet 3
#NO_CHANNEL	*No Channel*
#B1_CHANNEL	*B1 Channel*
#B2_CHANNEL	*B2 Channel*
#ANY_CHANNEL	*Any Channel*

1.1.6. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->PI_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->PI_LOCATION #USER #LOCAL_PRIVAT #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #REMOTE_PRIVATE #BEYOND_INTERWORK	Location, Octet 3 *User* *Local Private Nertwork* *Local Public Network* *Transit Network* *Remote Public Network* *Remote Private Network* *Network Beyond Interworking Point*
->PI_DESCRIPTION #NOT_END_TO_END #DEST_NON_ISDN #ORIG_NON_ISDN #RETURNED_TO_ISDN #INBAND_INFO_AVAIL	Progress Description, Octet 4 *Call is not end-to-end ISDN* *Destination Address is non-ISDN* *Origination Address is non-ISDN* *Call has Returned to the ISDN* *In-band Info now Available*

1.1.7. Terminal Capabilities IE (I#TERM_CAPAB)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->TC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->TC_DESCRIPTION #TYPE2_STIMULUS	Capability Description, Octet 3 *Type 2 stimulus mode terminal*

1.1.8. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.9. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#CUSTOM_ON	*Custom tone on*
#RECALL_DIAL_ON	*Recall dial tone on*
#BUSY_VERIFY_ON	*Busy verify tone on*
#ERROR_ON	*Error tone on*
#STUTTER_DIAL_ON	*Stutter dial tone on*
#EXPENSIVE_ROUTING_ON	*Expensive routing tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*
#UNSPEC_TONE	*Unspecified tone*

6.1.10. Switchhook IE (I#SWITCHHOOK)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SW_VALUE	Switchhook Value, Octet 3
#ON_HOOK	*On-hook*
#OFF_HOOK	*Off-hook*

1.1.11. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	AT&T Reference
->CLGN_NUMBER_TYPE	Type of Number, Octet 3
#UNKNOWN	*Unknown*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#SUBSCRIBER	*Subscriber Number*
#ABBREVIATED	*Abbreviated Number*
->CLGN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#UNKNOWN_PLAN	*Unknown*
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#PRIVATE_PLAN	*Private Numbering Plan*
->CLGN_PRESENTATION	Presentation Indicator, Octet 3a
#PRESENT_ALLOWED	*Presentation Allowed*
#PRESENT_RESTRICTED	*Presentation Restricted*
#NUMBER_UNAVAIL	*Number Not Available Due to Interworking*
->CLGN_SCREENING	Screening indicator, Octet 3a
#UNSCREENED	*User-Provided, Not Screened*
#VERIFY_PASSED	*User-Provided, Verified and Passed*
#VERIFY_FAILED	*User-Provided, Verified and Failed*
#NETWORK_PROVIDED	*Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

1.1.12. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->CLDN_NUMBER_TYPE	Type of Number, Octet 3
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#LOCAL_DIRECTORY	*Local Directory Number*
->CLDN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#PRIVATE_PLAN	*Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.13. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->RI_CLASS #INDICATED_CHANNEL #SINGLE_INTERFACE	Class of Channel or Interface, Octet 3 *Indicated B-Channel* *Single Interface*

6.1.14. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate, Octet 4 *Used for packet mode* *64 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC #BIDIRECT_ASYMMETRIC #UNIDIRECT_ORIG_DEST #UNIDIRECT_DEST_ORIG	Symmetry, Octet 4b *Bidirectional Symmetric* *Bidirectional Asymmetric* *Unidirectional orig -> dest* *Unidirectional dest -> orig*

Selector Name and related Constants	AT&T Reference
->LL_TRANSFER_RATE_4B #CHANNEL_SIZE #64KBIT/S	Information Transfer Rate (D->O), Octet 4b *Used for packet mode* *64 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->LL_L1_PROTOCOL #RATE_ADAPTION_V110/X30 #G.711_ULAW #G.711_ALAW #G.721_ADPCM #NON_CCITT	Layer 1 Protocol Identifier, Octet 5 *Rate Adaption V.110 / X.30* *Recommendation G.711 u-law* *Recommendation G.711 A-law* *Recommendation G.721 32 kbit/s ADPCM* *Non-CCITT rate adaption*
->LL_SYNC/ASYNC #SYNCHRONOUS #ASYNCHRONOUS	Synchronous/Asynchronous, Octet 5a *Synchronous* *Asynchronous*
->LL_USER_RATE #UNDEFINED #0.6KBIT/S #1.2KBIT/S #2.4KBIT/S #3.6KBIT/S #4.8KBIT/S #7.2KBIT/S #8KBIT/S #9.6KBIT/S #14.4KBIT/S #16KBIT/S #19.2KBIT/S #32KBIT/S #48KBIT/S #56KBIT/S	User rate, Octet 5a *Undefined* *0.6kbit/s Recommendation V.6 and X.1* *1.2kbit/s Recommendation V.6* *2.4kbit/s Recommendation V.6* *3.6kbit/s Recommendation V.6* *4.8kbit/s Recommendation V.6 and X.1* *7.2kbit/s Recommendation V.6* *8kbit/s Recommendation I.460* *9.6kbit/s Recommendation V.6 and X.1* *14.4kbit/s Recommendation V.6* *16kbit/s Recommendation I.460* *19.2kbit/s Recommendation V.6* *32kbit/s Recommendation I.460* *48kbit/s Recommendation V.6 and X.1* *56kbit/s Recommendation V.6*
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #Q.921 #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation Q.921 (i.441)* *Recommendation X.25, Link level*
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer*

1.2. CODESET 6 IES

1.2.1. User Code IE (I#USER_CODE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->UC_TYPE	Type of User Code, Octet 3
#ANY_CODE	*Any user code*
#ACCOUNT_CODE	*Account code*
#LOGIN_DIGITS	*Login digits*
#SUBSCRIBER_ID	*Subsciber id*
#AUTHOR_CODE	*Authorization code*
->UC_VALUE	User Code Value, Octet 4 *

1.2.2. Associated Type IE (I#ASSOC_TYPE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->AT_STATUS_VALUE	Status Value, Octet 3
#SETUP	*Setup*
#CONNECT	*Connect*
#HOLD	*Hold*
#RECONNECT	*Reconnect*
#EXCLUSION	*Exclusion*
#CONNECT_DENIED	*Connect denied*
#CLEAR_DENIED	*Clearing denied*

1.2.3. Selected Call Appearance IE (I#SELECT_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->SCA_BUTTON_NUM	Button Number, Octet 3

6.2.4. Origination Call Appearance IE (I#ORIG_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->OCA_VALUE	Origination Call Appearance Value, Octet 3

6.2.5. Destination Call Appearance IE (I#DEST_CALL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->DCA_VALUE	Destination Call Appearance Value, Octet 3

6.2.6. Endpoint Identifier IE (I#ENDPOINT_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->EID_USER_SERVICE_ID	User Service Identifier, Octet 3
->EID_INTERPRETER	Interpreter, Octet 4
->EID_TERMINAL_ID	Terminal Identifier, Octet 4

6.2.7. Keypad Control IE (I#KEYPAD_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->KC_CR_VALUE_TYPE #NULL_CALL_REF #NON_NULL_CALL_REF	Call Reference Value Type, Octet 3 *Null call reference* *Non-null call reference*

6.2.8. Other Call Reference IE (I#OTHER_CR)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->OCR_FLAG #ORIG_SIDE #DEST_SIDE	Flag, Octet 3 *Origination side* *Destination side*
->OCR_VALUE	Other Call Reference Value, Octet 3

6.2.9. Feature Activation IE (I#FEATURE_ACT)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	AT&T Reference
->FA_BUTTON_TYPE #UNKNOWN #CALL_APPEARANCE #FEATURE_BUTTON	Button Type, Octet 3 *Unknown* *Call appearance button* *Feature button*
->FA_MODULE_NUM	Module Number, Octet 3
->FA_STATUS_TYPE	Status Type, Octet 3
->FA_BUTTON_NUM	Button Number, Octet 4

6.2.10. Feature Indication IE (I#FEATURE_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7

Selector Name and related Constants	AT&T Reference
->FI_BUTTON_TYPE #UNKNOWN #CALL_APPEARANCE #FEATURE_BUTTON #NONE_APPLICABLE	Button Type, Octet 3 *Unknown* *Call appearance button* *Feature button* *None applicable*
->FI_MODULE_NUM	Module Number, Octet 3
->FI_STATUS_TYPE #FEATURE_NUMBER_STATUS #FEATURE_BUTTON_STATUS #MULTILE_BUTTON_STATUS #MAINTENANCE_STATUS	Status Type, Octet 3 *Feature number status* *Feature button status* *Multiple button status* *Maintenance status*

6.2.10.1. Further Octets for Status Type: #FEATURE_NUMBER_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Number, Octet 4
->FI_STATUS #ACTIVATED #DEACTIVATED #PENDING #LOCAL_HOLD #REMOTE_HOLD #CONFIRMED #REQUESTED_STATE #REJECTED	Status, Octet 5 *Activated* *Deactivated* *Pending* *Local hold* *Remote hold* *Confirmed* *Already in requested state* *Rejected*
->FI_ADD_INFO	Additional Information, Octet 6 * (for repetition of Octet 4 and Octet 5)

1.2.10.2. Further Octets for Status Type: #FEATURE_BUTTON_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Number, Octet 4
->FI_INDICATOR	Indicator, Octet 5
->FI_STATUS #ACTIVATED #DEACTIVATED #PENDING #LOCAL_HOLD #REMOTE_HOLD #CONFIRMED #REQUESTED_STATE #REJECTED	Status, Octet 5 *Activated* *Deactivated* *Pending* *Local hold* *Remote hold* *Confirmed* *Already in requested state* *Rejected*
->FI_ADD_INFO	Additional Information, Octet 6 * (for repetition of Octet 4 and Octet 5)

1.2.10.3. Further Octets for Status Type: #MULTIPLE_BUTTON_STATUS

Selector Name and related Constants	AT&T Reference
->FI_NUMBER	Feature Button Number, Octet 4
->FI_NUM_BUTTONS	Number of Buttons, Octet 5
->FI_STATUS	Status, Octet 6
->FI_ADD_INFO	Additional Information, Octet 7 * (for more Status Information)

1.2.10.4. Further Octets for Status Type: #MAINTENANCE_STATUS

Selector Name and related Constants	AT&T Reference
->FI_MAINTENANCE #ALL_INDICATORS_OFF	Maintenance, Octet 4 *Turn all indicators off*

1.2.10.5. Further Octets for other Status Type

Selector Name and related Constants	AT&T Reference
->FI_ADD_INFO	Additional Information, Octet 4 *

6.2.11. Adjunct Control IE (I#ADJUNCT_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->AC_VALUE	Adjunct Control Value, Octet 3
#OFF	*Off*
#ON	*On*

6.2.12. Display Control IE (I#DISPLAY_CTRL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	AT&T Reference
->DC_MODE	Display mode, Octet 3
#NO_CHANGE	*No change*
#NORMAL	*Normal*
#INSPECT	*Inspect*
#MISC_DISPLAY_INFO	*Misc. display information*
#MSG_RETRIEVAL	*Message retrieval*
#ELECTRONIC_DIRECTORY	*Electronic directory query*

6.2.13. Display Field IE (I#DISPLAY_FLD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	AT&T Reference
->DF_MODE #NORMAL #INSPECT #MISC_DISPLAY_INFO #MSG_RETRIEVAL #ELECTRONIC_DIRECTORY	Display Mode, Octet 3 *Normal* *Inspect* *Misc. display information* *Message retrieval* *Electronic directory query*
->DF_SUBMODE #NO_SUBMODE #DIRECT #REDIRECTED	Submode, Octet 3 *No submode applicable* *Direct* *Redirected*
->DF_TYPE #CALL_APPEAR_ID #CALLED_PARTY_ID #CALLING_PARTY_ID #CALLED_PARTY_NAME #CALLING_PARTY_NAME #ORIG_PERMISSION #ISDN_CALL_ID #MISC_CALL_INFO #ENTIRE_DISPLAY #DATE_AND_TIME	Display Field Type, Octet 4 *Call appearance ID* *Called party ID* *Calling party ID* *Called party name* *Calling party name* *Originating permission* *ISDN call identification* *Misc. call information* *Entire display* *Date and time of day*
->DF_INFO	Display Information, Octet 5 *

6.2.14. Management IE (I#MANAGEMENT)

Possible octet inclusions/exclusions:

OCTET_7C, OCTET_7D, OCTET_7E, OCTET_7F, OCTET_7G,
OCTET_7H, OCTET_7I, OCTET_7I.1, OCTET_7I.2

Selector Name and related Constants	AT&T Reference
->M_PROTOCOL_DISCR	Protocol Discriminator, Octet 7c
#AT&T_ISDN_PROTOCOL	*AT&T ISDN management protocol*
->M_TRANSACTION_REF	Transaction Reference, Octet 7d
->M_OP_CLASS	Operation Class, Octet 7e
#UNCONFIRMED_OPERATION	*Unconfirmed operation*
#CONFIRMED_OPERATION	*Confirmed operation*
#RETURN_RESULT	*Return result*
#RETURN_RESULT_CONT	*Return result continued*
#RETURN_ERROR	*Return error*
#REJECT	*Reject*
->M_OP_TYPE	Operation Type, Octet 7e
#ACTION	*Action*
#EVENT_REPORT	*Event report*
#GET	*Get*
#SET	*Set*
->M_CODE	Code, Octet 7f
for Operation Type #ACTION	
#ACTIVATE_LOOPBACK	*Activate loopback*
#DEACTIVATE_LOOPBACK	*Deactivate loopback*
#RESET	*Reset*
for Operation Type #EVENT_REPORT	
#INIT_REQUEST	*Initialization request*
#ENDPOINT_SERV_CHANGE	*Endpoint service state change*
for Operation Type #GET	
#SERVICE_PROFILE_INFO	*Service profile information*
#CALL_STATUS_INFO	*Call status information*
#EQUIPMENT_INFO	*Equipment information*
for Operation Type #SET	
#SERVICE_PROFILE_INFO	*Service profile information*
#ADDRESS_INFO	*Address information*
for Operation Class #REJECT	
#UNSPEC_ERROR	*Unspecified error*
#PROTOCOL_VIOLATION	*Protocol violation*
#UNREC_OPERATION	*Unrecognized operation*
#NON_SYNC_OPERATION	*Non-synchronous operation not supported*

Selector Name and related Constants	AT&T Reference
->M_PARAMETER_ID #SERVICE_PROFILE_ID #ENDPOINT_ID #CALL_REF_ID #CALL_STATE_ID #CHANNEL_ID #SERVICE_STATE #SERVICE_MSG_ERROR #LOOPBACK_LOCATION #SUPPL_CAPABILITIES #CAUSE_ID	Parameter Identifier, Octet 7g *Service profile identifier* *Endpoint identifier* *Call reference identifier* *Call state identifier* *Channel identifier* *Service state* *Service message error code* *Loopback location* *Supplementary capabilities* *Cause identifier*
->M_PARAMETER_LEN	Length of Paremeter, Octet 7h

6.2.14.1. Parameter Field for Parameter Identifier: #SERVICE_PROFILE_ID

Selector Name and related Constants	AT&T Reference
->M_SPID_VALUE	SPID value, Octet 7i *

6.2.14.2. Parameter Field for Parameter Identifier: #ENDPOINT_ID

Selector Name and related Constants	AT&T Reference
->M_USID	USID, Octet 7i.1
->M_FLAG #RESTRICTED_SERVICE #FULL_SERVICE	Flag, Octet 7i.2 *Restricted service* *Full service*
->M_TID	TID, Octet 7i.2

6.2.14.3. Parameter Field for Parameter Identifier: #CALL_REF_ID

Selector Name and related Constants	AT&T Reference
->M_FLAG #ORIG_SIDE #DEST_SIDE	Flag, Octet 7i *Origination side* *Destination side*
->M_CALL_REF	Call reference value, Octet 7i

6.2.14.4. Parameter Field for Parameter Identifier: #CALL_STATE_ID

Selector Name and related Constants	AT&T Reference
->M_CALL_STATE	Call State, Octet 7i
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_RECEIVED	*Call Request*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#RELEASE_REQUEST	*Release Request*

6.2.14.5. Parameter Field for Parameter Identifier: #CHANNEL_ID

Selector Name and related Constants	AT&T Reference
->M_INT_PRESENT	Interface id. present, Octet 7i
#IMPLICIT	*Implicitly identifier*
->M_INT_TYPE	Interface Type, Octet 7i
#BASIC_INTERFACE	*Basic Interface*
->M_PREF / EXCL	Preferred / Exclusive, Octet 7i
#PREFERRED	*Preferred*
#EXCLUSIVE	*Exclusive*
->M_DCHANNEL	D-Channel Idicator, Octet 7i
#NOT_D_CHANNEL	*Not D-Channel*
#D_CHANNEL	*D-Channel*
->M_INFO_CHAN_SEL	Info Channel Selection, Octet 7i
#NO_CHANNEL	*No Channel*
#B1_CHANNEL	*B1 Channel*
#B2_CHANNEL	*B2 Channel*
#ANY_CHANNEL	*Any Channel*

6.2.14.6. Parameter Field for Parameter Identifier: #SERVICE_STATE

Selector Name and related Constants	AT&T Reference
->M_SERVICE_STATE	Service State, Octet 7i
#IN_SERVICE	*In service*
#OUT_OF_SERVICE	*Out of service*

1.2.14.7. Parameter Field for Parameter Identifier: #SERVICE_MSG_ERROR

Selector Name and related Constants	AT&T Reference
->M_ERROR_CODE #TEST_IN_PROGRESS	Error Code, Octet 7i *Test in Progress*

1.2.14.8. Parameter Field for Parameter Identifier: #LOOPBACK_LOCATION

Selector Name and related Constants	AT&T Reference
->M_LOOPBACK	Loopback Location, Octet 7i *

1.2.14.9. Parameter Field for Parameter Identifier: #SUPPL_CAPABILITIES

Selector Name and related Constants	AT&T Reference
->M_DISPLAY #SUPPORTED #NOT_SUPPORTED	Display, Octet 7i.1 *Display capabilities supported* *Display capabilities not supported*
->M_SUPPL_DATA #SUPPORTED #NOT_SUPPORTED	Suppl. Data, Octet 7i.1 *Supplementary data services supported* *Supplementary data services not supported*
->M_FC #SUPPORTED #NOT_SUPPORTED	FC, Octet 7i.1 *Flexible call offering supported* *Flexible call offering not supported*
->M_TMF #SUPPORTED #NOT_SUPPORTED	TMF, Octet 7i.1 *Terminal management feature supported* *Terminal manag. feature not supported*
->M_ASSOC #SUPPORTED #NOT_SUPPORTED	ASSOC, Octet 7i.1 *Key system protocol supported* *Key system protocol not supported*
->M_LOOP #SUPPORTED #NOT_SUPPORTED	Loop, Octet 7i.2 *Loopback capabilities supported* *Loopback capabilities not supported*
->M_SVC #SUPPORTED #NOT_SUPPORTED	Svc., Octet 7i.2 *Endpoint service msg procedures supported* *Endpoint service msg procedures not supp.*
->M_INIT #SUPPORTED #NOT_SUPPORTED	Init., Octet 7i.2 *Endpoint initialization supported* *Endpoint initialization not supported*

1.2.14.10. Parameter Field for Parameter Identifier: #CAUSE_ID

Selector Name and related Constants	AT&T Reference
->M_CAUSE	Cause Value, Octet 7i
#CHANNEL_IN_USE	*Channel in use*
#MNG_PARAM_UNKNOWN	*Provided management parameter unknown*
#INVALID_PARAM	*Invalid parameter value*

1.2.14.11. Parameter Field for Other Parameter Identifier

Selector Name and related Constants	AT&T Reference
->M_PARAMETER_VAL	Parameter Value, Octet 7i *

Note that only **one** Parameter may be coded/decoded in a Management IE.

ISDN LAYER 3 NATIONAL VARIANT (1TR6)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.1 Rev.0
System Version	:	MF 2.2 Rev. 1
Revision Date	:	January 1993
Engineering	:	AUT E33 D / SteHu
Quality Control	:	MWB KQP 2 HBR

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

2.1. MESSAGES WITH PD N1 2-1

2.2. MESSAGES WITH PD N0 2-2

3. W-ELEMENT IDENTIFIER 3-1

3.1. CODESET 0 3-1

3.2. CODESET 6 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING W-ELEMENT FIELD VALUES 6-1

6.1. NATIONAL CS 0 IES (W-ELEMENTS) 6-1

6.1.1. SHIFT (I#SHIFT) 6-1

6.1.2. CONGESTION LEVEL (I#CONG_LEVEL) 6-1

6.1.3. CAUSE (I#CAU) 6-2

6.1.4. CONNECTED ADDRESS (I#CAD) 6-3

6.1.5. CALL IDENTITY (I#CAI) 6-3

6.1.6. CHANNEL ID (I#CHI) 6-4

6.1.7. NETWORK SPECIFIC FACILITIES (I#NSF) 6-5

6.1.7.1. PARAMETER FIELD FOR FACILITY CODE #SPERRE 6-8

6.1.7.2. PARAMETER FIELD FOR FAC. CODE #UMLEITUNG_STAENDIG/#UMLEITUNG.FALLWEISE 6-8

6.1.7.3. PARAMETER FIELD FOR FAC. CODE #B_KANAL_UEBERNEHMEN 6-9

6.1.7.4. PARAMETER FIELD FOR FAC. CODE #DIENSTWECHSEL_1/#DIENSTWECHSEL_2 6-9

6.1.7.5. PARAMETER FIELD FOR FAC. CODE #GBG 6-9

6.1.8. DISPLAY (I#DSP) 6-10

6.1.9. KEYPAD (I#KEY) 6-10

6.1.10. ORGINATION ADDRESS (I#OAD) 6-10

6.1.11. DESTINATION ADDRESS (I#DAD) 6-11

6.1.12. USER-USER INFORMATION (I#UUI) 6-11

6.2. NATIONAL CS 6 IES (W-ELEMENTS) 6-12

6.2.1. SERVICE INDICATOR (I#SIN) 6-12

6.2.2. CHARGING INFO (I#CIF) 6-14

6.2.3. DATE (I#DTE) 6-14

6.2.4. FACILITY SELECT (I#FSE) 6-15

6.2.5. STATUS OF FACILITIES (I#SOF) 6-15

6.2.6. STATUS DES GERUFENEN TLN (I#SDT) 6-16

6.2.7. ADDITIONAL TRANSMISSION ATTRIBUTES (I#ATA) 6-16

Program:	1TR6 (National variant of the ISDN layer 3 protocol)
Software Version:	V3.1
Previous Version:	V3.0
System Version:	MF 2.2.1
Revision Date:	January 1993

☞ *Please note that it is important to read this paper before reading the manual, because some changed functions are described there.*

CHANGES MADE:

- Updated up to 'Berichtigung Nr. 7' of the 1TR6 protocol.

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

2.1. MESSAGES WITH PD N1

Message Identifier	1TR6 Reference
M#ALERT	Alerting
M#CALL_SENT	Call Sent
M#CONN	Connect
M#CONN_ACK	Connect Acknowledge
M#SETUP	Setup
M#SETUP_ACK	Setup Acknowledge
M#RES	Resume
M#RES_ACK	Resume Acknowledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknowledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DET	Detach
M#DISC	Disconnected
M#REL	Release
M#REL_ACK	Release Acknowledge
M#CANC_ACK	Cancel Acknowledge
M#CANC_REJ	Cancel Reject
M#CON_CON	Congestion Control
M#FAC	Facility
M#FAC_ACK	Facility Acknowledge
M#FAC_CANC	Facility Cancel
M#FAC_REG	Facility Register
M#FAC_REJ	Facility Reject
M#INFO	Information
M#REG_ACK	Register Acknowledge
M#REG_REJ	Register Reject
M#STATUS	Status

2.2. MESSAGES WITH PD N0

Message Identifier	1 TR6 Reference
M#REG_IND	Register Indication
M#CANC_IND	Cancel Indication
M#FAC_STA	Facility Status
M#STA_ACK	Status Acknowledge
M#STA_REJ	Status Reject
M#FAC_INF	Facility Information
M#INF_ACK	Information Acknowledge
M#INF_REJ	Information Reject
M#CLOSE	Close
M#CLO_ACK	Close Acknowledge

3. W-ELEMENT IDENTIFIER

3.1. CODESET 0

W-Element	1TR6 Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#CONG_LEVEL	Congestion Level
I#CAU	Cause
I#CAD	Connected Address
I#CAI	Call Identity
I#CHI	Channel Identification
I#NSF	Network Specific Facilities
I#DSP	Display
I#KEY	Keypad
I#OAD	Orgination Address
I#DAD	Destination Address
I#UUI	User-User Information

3.2. CODESET 6

W-Element	1TR6 Reference
I#SIN	Service Indicator
I#CIF	Charging Information
I#DTE	Date
I#FSE	Facility Select
I#SOF	Status of Facility
I#SDT	Status of the called address
I#ATA	Additional transmission attributes

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements ("W-Elements") :

OCTET_3, OCTET_4, OCTET_4.1 OCTE_4.2 OCTET_5,
OCTET_6, OCTET_7, OCTET_8, OCTET_9, OCTET_10
OCTET_12, OCTET_13, OCTET_14.

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds) network side	user side P-MP	user side P-P
Timer T302	#T302	12	-	12
Timer T303	#T303	1,5	4	4
Timer T304	#T304	-	20	60
Timer T305	#T305	120	4	30
Timer T307	#T307	120	-	-
Timer T308	#T308	4	4	4
Timer T310	#T310	-	120	70
Timer T313	#T313	-	4	4
Timer T318	#T318	-	4	-
Timer T319	#T319	-	4	-
Timer T3AA	#T3AA	120	-	120
Timer T3AB	#T3AB	15	-	-
Timer T3AC	#T3AC	20	-	-
Timer T3AD	#T3AD	30	-	-
Timer T3AE	#T3AE	180	-	-
Timer T3AF	#T3AF	15 min	-	-
Timer T3AG	#T3AG	5	-	-
Timer T3D1	#T3D1	-	10	10
Timer T3D2	#T3D2	-	4	4

6. ACCESSING W-ELEMENT FIELD VALUES

The following sections contain all information for access of the Information Element parameter list including relevant octet identifier for all Information Elements,selector names associated with each Information Element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. NATIONAL CS O IES (W-ELEMENTS)

6.1.1. SHIFT (I#SHIFT)

Selector Name and related Constants	1 TR6 Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	New codeset identification
#CODESET0	*Codesatz 0*
#CODESET6	*Codesatz 6*
#CODESET7	*Codesatz 7*

6.1.2. CONGESTION LEVEL (I#CONG_LEVEL)

Selector Name and related Constants	1 TR6 Reference
->CL_CONGESTION_LEVEL	Congestion Level
#RECEIVER_READY	*Receiver Ready*
#RECEIVER_NOT_READY	*Receiver not Ready*

6.1.3. CAUSE (I#CAU)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 REFERENCE
->CAU_VALUE	Cause value, Octet 3
1TR6#INVALID_CR	*Invalid call reference value*
1TR6#BEARER_SERVICE_UNIMPL	*Bearer service not implemented*
1TR6#CALL_ID_NONEXISTENT	*Call identity does not exist*
1TR6#CALL_ID_IN_USE	*Call identity in use*
1TR6#NO_CHANNEL_AVAIL	*No Channel available*
#FACILITY_UNIMPL	*Requested facility not implemented*
#FACILITY_NOT_SUBSCRIBED	*Requested facility not subscribed*
#OUTGOING_CALL_BARRED	*Outgoing calls barred*
#USER_ACCESS_BUSY	*User access busy*
#NEG_GBG_VGL	*Connection not possible (neg. CUG comp.)*
#KOM_SPV_NOTALLOWED	*Connection not possible (neg. call number check)*
#NOT_OBTAINABLE	*Destination not obtainable*
1TR6#NUMBER_CHANGED	*Number changed*
1TR6#OUT_OF_ORDER	*Out of order*
#NO_RESPONSE	*No user responding*
1TR6#USER_BUSY	*User busy*
#INCOMING_BARRED	*Incoming calls barred*
1TR6#CALL_REJECTED	*Call rejected*
#NTWK_CONGESTION	*Network congestion*
#REMOTE_INITIATED	*Remote user initiated*
#LOC_PROC_ERR	*Local procedure error*
#REM_PROC_ERR	*Remote procedure error*
#REM_SUSPENDED	*Remote user suspended*
#REM_RESUMED	*Remote user resumed*
#USER_INFO_DISCARDED	*User info discarded locally*
#NONEXISTENT_CUG	*Non existent CUG*
->CAU_CODING_STANDARD	Coding standard, Octet 4
#NATIONAL	*National standard*
->CAU_LOCATION	Location, Octet 4
#PRIVATE	*Private network*
#PUBLIC	*Public network*
#NO_INFO	*No information*

6.1.4. CONNECTED ADDRESS (I#CAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->CAD_ADDR_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL	Type of address, Octet 3 *Unknown* *International Number* *National Number*
->CAD_ADDR_PLAN #UNKNOWN_PLAN #ISDN_PLAN	Numbering plan ID, Octet 3 *Unknown* *ISDN Numbering Plan E. 164*
->CAD_ADDRESS	Address (max. length 16 Bytes) Octet 4 *

6.1.5. CALL IDENTITY (I#CAI)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->CAI_ID (any pattern allowed)	Call identity, Octet 3

6.1.6. CHANNEL ID (I#CHI)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	1 TR6 Reference
->CHI_PAT1 **#CHI_BA** **#CHI_PRI**	Bits 5-8 in Octet 3 : *Valid pattern for Basic Access* *Valid pattern for Primary Rate*
->CHI_P/E **#PREFERRED** **#EXCLUSIVE**	Preferred / exclusive bit, Octet 3 *Preferred* *Exclusive*
->CHI_PAT2 **#CHI_PAT2**	Bit 3 in Octet 3 *Valid pattern bit 3 in octet 3*
->CHI_ICS BASIC ACCESS: **#NO_CHANNEL** **#B1_CHANNEL** **#B2_CHANNEL** **#ANY_CHANNEL** PRIMARY RATE : **#NO_CHANNEL** **#SEE_CHAN_NO** **#ANY_CHANNEL**	Info Channel Selection, Octet 3 *Kein Kanal* *B1 Kanal* *B2 Kanal* *Beliebiger Kanal* *Kein Kanal* *Siehe "Channel Number"* *Beliebiger Kanal*
->CHI_PAT3 **#CHI_PAT3**	Bits 1 - 8 in Octet 4 (PRI only) *Valid pattern*
->CHI_CHANNEL	Channel number, Octet 5 (PRI only)

6.1.7. NETWORK SPECIFIC FACILITIES (I#NSF)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8, OCTET_9, OCTET_10, OCTET_12, OCTET_13, OCTET_14

Selector Name and related Constants	1 TR6 Reference
->NSF_PAT	Bits 1 - 8 in Octet 3
#NSF_PAT	*Valid pattern*
->NSF_CODE	facility code, Octet 4
#SPERRE	*Sperre*
#UMLEITUNG_STAENDIG	*Anrufumleitung (ständig)*
#UMLEITUNG_FALLWEISE	*Anrufumleitung (fallweise)*
#KONFERENZ	*Konferenz*
#B_KANAL_UEBERNEHMEN	*B-Kanal übernehmen*
#HALT_AKTIVIEREN	*Aktivieren einer gehaltenen Verbindung*
#DREIERKONFERENZ	*Dreierkonferenz*
#DIENSTWECHSEL_1	*einseitiger Dienstwechsel*
#DIENSTWECHSEL_2	*zweiseitiger Dienstwechsel*
#NR_IDENTIFIZIERUNG	*Rufnummer-Identifizierung*
#GBG	*Geschlossene Benutzergruppe*
#ANZ_UEBERGEBENER_RUF	*Anzeige: übergebener Ruf*
#ANZ_UMLEITUNG	*Anzeige: umgel./weitergesch. Ruf*
#ANZ_UNTERDRUECKEN	*Unterdrücken A-Rufnummer*
#VERB_DEAKTIVIEREN	*Nutzung d. Verbindung deaktivieren*
#VERB_AKTIVIEREN	*Nutzung d. Verbindung aktivieren*
#SEMIPERMANENT	*Semipermanente Verbindung (SPV)*
#RUECKWECHSEL	*Rückwechsel bei zweiseit. D-Wechsel*
#UMLEITUNG_PRIVAT	*Anrufumleitung im privaten Netz*

->NSF_SERVICE	Service, Octet 5
#UNKNOWN	*Gilt für alle Dienste / irrelevant*
#FERNSPRECHEN	*Fernsprechen*
->NSF_ADD_INFO	Additional Information, Octet 6
#ISDN_3.1KHZ	*ISDN-Fernsprechen 3,1 kHz*
#ANALOG	*Fernsprechen analog*
#ISDN_7KHZ	*ISDN-Fernsprechen 7 kHz*
#A/B-DIENSTE	*a/b-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#FAX_GRUPPE_2	*Fax. Gruppe 2*
#FAX_GRUPPE_3	*Fax. Gruppe 3*
#MODEM_DATEN	*Daten über Modem*
#BTX_MODEM	*Btx über Btx-Modem*
#X.21-DIENSTE	*X.21-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#UC_4	*User Class 4*
#UC_5	*User Class 5*
#UC_6	*User Class 6*
#UC_19	*User Class 19*
#FAX_GRUPPE_4	*Telefax Gruppe 4*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BTX_64KBIT/S	*Btx 64 kBit/s*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#DUE_64KBIT/S	*Datenübertragung 64 kBit/s*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
	Nur relevant bei Facility-Code Zweiseitiger/Einseitiger Dienstwechsel und wird dann codiert wie "Additional Information" in "Service Indicator"
#X.25-DIENSTE	*X.25-Dienste*
->NSF_ADD_INFO	Additional Information, Octet 6
#UC_8	*User Class 8*
#UC_9	*User Class 9*
#UC_10	*User Class 10*
#UC_11	*User Class 11*
#UC_13	*User Class 13*
#19.2KBIT/S	*19,2 kbit/s*

#TELETEX_64KBIT/S	*Teletex 64*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#MIXED_MODE_1TR6	*Mixed Mode*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#FERNWIRKEN	*Fernwirken*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#GRAFIKTELEFON	*Grafiktelefondienste*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BILDSCHIRMTEXT	*Bildschirmtext (neuer Standard)*
->NSF_ADD_INFO	Additional Information, Octet 6
N#ADD_INFO	*nicht relevant / default (0)*
#BILDTELEFON	*Bildtelefon*
->NSF_ADD_INFO	Additional Information, Octet 6
#TON_3.1KHZ	*Ton 3,1 kHz*
#TON_7KHZ	*Ton 7 kHz*
#BILD	*Bild*

6.1.7.1. PARAMETER FIELD FOR FACILITY CODE #SPERRE

Selector Name and related Constants	1 TR6 Reference
->NSF_AS #SPERR_ALLE #SPERR_FERN #SPERR_AUSL #SPERR_INTERKONT	Art der Sperre, Octet 7 *Sperre aller abgehender Verbindungen* *Sperre abghd. Fernverbindungen* *Sperre abghd. Auslandsverbindungen* *Sperre abghd. Interkontinental-verbindungen*
->NSF_AS2 **Parameter wie bei "->NSF_AS"**	Art der Sperre kann mehrfach auftreten, Octet 8
->NSF_AS3 **Parameter wie bei "->NSF_AS"**	Art der Sperre, Octet 9
->NSF_AS4 **Parameter wie bei "->NSF_AS"**	Art der Sperre, Octet 10

6.1.7.2. PARAMETER FIELD FOR FAC. CODE #UMLEITUNG_STAENDIG / #UMLEITUNG.FALLWEISE

Selector Name and related Constants	1 TR6 Reference
->NSF_ADDR_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL	Address number type, Octet 7 *Unknown* *International Number* *National Number*
->NSF_ADDR_PLAN #UNKNOWN_PLAN #ISDN_PLAN	Address numbering plan, Octet 7 *Unknown* *ISDN Numbering Plan E.164*
->NSF_ADDRESS	Address, Octet 8 * (max. length 16 Bytes)

6.1.7.3. PARAMETER FIELD FOR FAC. CODE #B_KANAL_UEBERNEHMEN

Selector Name and related Constants	1 TR6 Reference
->NSF_CHANNEL	B-Kanalnummer, Octet 7
Basisanschluß:	
#B1_CHANNEL	*B1 Kanal*
#B2_CHANNEL	*B2 Kanal*
Primärmultiplexanschluß:	
wie "channel number" in "Channel Identification"	

6.1.7.4. PARAMETER FIELD FOR FAC. CODE #DIENSTWECHSEL_1 / #DIENSTWECHSEL_2

Selector Name and related Constants	1 TR6 Reference
->NSF_EAZ	EAZ der initiierenden Seite, Octet 7 * (length: 1 Byte)
->NSF_EAZ_ABH	EAZ der abhängigen Seite Octet 8 * (length: 1 Byte)

6.1.7.5. PARAMETER FIELD FOR FAC. CODE #GBG

Selector Name and related Constants	1 TR6 Reference
->NSF_GBG	GBG, Octet 7 * (length: 2 Bytes)

6.1.8. DISPLAY (I#DSP)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->DSP_DISPLAY	Octet 3 * , (max. length: 40 Bytes)

6.1.9. KEYPAD (I#KEY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->K_KEYPAD	Keypad, Octet 3 * (max. length: 32 Bytes)

6.1.10. ORGINATION ADDRESS (I#OAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->OAD_ADDR_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL	Address number type, Octet 3 *Unknown* *International Number* *National Number*
->OAD_ADDR_PLAN #UNKNOWN_PLAN #ISDN_PLAN	Address numbering plan, Octet 3 *Unknown* *ISDN Numbering PLan E. 164*
->OAD_ADDRESS	Address Octet 4 * (max. length 16 Bytes)

6.1.11. DESTINATION ADDRESS (I#DAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Reference
->DAD_ADDR_TYPE **#UNKNOWN** **#INTERNATIONAL** **#NATIONAL**	Address number type, Octet 3 *Unknown* *International Number* *National Number*
->DAD_ADDR_PLAN **#UNKNOWN_PLAN** **#ISDN_PLAN**	Address numbering plan, Octet 3 *Unknown* *ISDN Numbering PLan E. 164*
->DAD_ADDRESS	Address Octet 4 * (max. length 16 Bytes)

6.1.12. USER-USER INFORMATION (I#UUI)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->UUI_INFO	User-User Information, Octet 3 * (max. length 255 Bytes)

6.2. NATIONAL CS 6 IES (W-ELEMENTS)

6.2.1. SERVICE INDICATOR (I#SIN)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4.1, OCTET_4.2

Selector Name and related Constants	1 TR6 Refernce
->SIN_SERVICE	Service, Octet 3
#FERNSPRECHEN	*Fernsprechen*
->SIN_ADD_INFO	Additional Info, Octet 4
#ISDN_3.1KHZ	*ISDN-Fernsprechen 3,1 kHz*
#ANALOG	*Fernsprechen analog*
#ISDN_7KHZ	*ISDN-Fernsprechen 7 kHZ*
#A/B-DIENSTE	*A/B-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#FAX_GRUPPE_2	*Fax, Gruppe 2*
#FAX_GRUPPE_3	*Fax, Gruppe 3*
#MODEM_DATEN	*Daten über Modem*
#BTX_MODEM	*Btx über Btx_Modem*
#X.21-DIENSTE	*X.21-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#UC_4	*User Class 4*
#UC_5	*User Class 5*
#UC_6	*User Class 6*
#UC_19	*User Class 19*
#FAX_GRUPPE_4	*Telefax Gruppe 4*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BTX_64KBIT/S	*Btx 64 kBit/s*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#DUE_64KBIT/S	*Datenübertragung 64 kBit/s*
Add. info octet f. sync. Übertragung:	
->SIN_SYNC_PAT	Synchrone bit pattern - OCTET_4.1
#SYNC_PAT	*Synchrone Übertragung mit Bitratenadapt.*
->SIN_SYNC_L1_PROT	Layer 1 Protocol - OCTET_4.1
#V110/X30_SYNC	*V.110, X.30 (ECMA 102)*
#V120	*V.120*

->SIN_SYNC_USR_RATE	User Rate - OCTET_4.1
#1200BIT/S	*1200 bit/s*
#1200/75BIT/S	*1200/75 bit/2*
#75/1200BIT/S	*75/1200 bit/s*
#2400BIT/S	*2400 bit/s*
#4800BIT/S	*4800 bit/s*
#9600BIT/S	*9600 bit/s*
#14400BIT/S	*14400 bit/s*
#19200BIT/S	*19200 bit/s*
#48000BIT/S	*48000 bit/s*
#56000BIT/S	*56000 bit/s*
#IN_BAND	*in-band negotiation*
Add. info octet f. async. Übertragung:	
->SIN_ASYNC_L1_PROT	Layer 1 Protocol - OCTET_4.2
#V110/X30_ASYNC	*V.110, X.30 (ECMA102)*
->SIN_DATA_BITS	Number of Databits - OCTET_4.2
#8_DATA_BITS	*8 data bits*
#7_DATA_BITS	*7 data bits*
->SIN_STOP_BITS	Number of Stopbits - OCTET_4.2
#1_STOP_BIT	*1 stop bit*
#2_STOP_BITS	*2 stop bits*
->SIN_PARITY	Parity - OCTET_4.2
#NO_PARITY	*No parity*
#EVEN_PARITY	*Even Parity*
->SIN_ASYNC_USR_RATE	User Rate - OCTET_4.2
#1200BIT/S	*1200 bit/s*
#1200/75BIT/S	*1200/75 bit/2*
#75/1200BIT/S	*75/1200 bit/s*
#2400BIT/S	*2400 bit/s*
#4800BIT/S	*4800 bit/s*
#9600BIT/S	*9600 bit/s*
#14400BIT/S	*14400 bit/s*
#19200BIT/S	*19200 bit/s*
#X.25-DIENSTE	*X.25-Dienste*
->SIN_ADD_INFO	Additional Info, Octet 4
#UC_8	*User Class 8*
#UC_9	*User Class 9*
#UC_10	*User Class 10*
#UC_11	*User Class 11*
#UC_13	*User Class 13*
#19.2KBIT/S	*19,2 kbit/s*
#TELETEX_64KBIT/S	*Teletex 64*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#MIXED_MODE_1TR6	*Mixed Mode*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*

#FERNWIRKEN	*Fernwirken*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#GRAFIKTELEFON	*Grafiktelefon*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BILDSCHIRMTEXT	*Bildschirmtext*
->SIN_ADD_INFO	Additional Info, Octet 4
N#ADD_INFO	*nicht relevant / default (0)*
#BILDTELEFON	*Bildtelefon*
->SIN_ADD_INFO	Additional Information, Octet 4
#TON_3.1KHZ	*Ton 3,1 kHz*
#TON_7KHZ	*Ton 7 kHz*
#BILD	*Bild*

6.2.2. CHARGING INFO (I#CIF)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	1 TR6 Refernece
->CIF_GEA #ANZAHL_EINHEITEN	Art der Gebühreninformation, Octet 3 *Anzahl der aufsummierten Gebühreninformat.*
->CIF_ANZAHL	IA5 codiert '0' -'9', Octet 4 * (max. length 12 Bytes)

6.2.3. DATE (I#DTE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Refernece
->DTE_DATE	Date: 'TT.MM.YY-HH:MM', Octet 3 * (length 14 Bytes)

6.2.4. FACILITY SELECT (I#FSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6

Selector Name and related Constants	1 TR6 Reference
->FSE_FAC_CODE	Facility Code, Octet 3 (coded as octet 4 of NSF)
->FSE_SERVICE	Service, Octet 4 (coded as octet 3 of Service Indicator)

6.2.5. STATUS OF FACILITIES (I#SOF)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 REFERNCE
->SOF_N_DM	Number of "aktive Dienstmerkmale"
->SOF_DM_01 ... **->SOF_DM_16** **#SPERRE_ABGEHEND** **#UMLEITUNG_STAENDIG** **#UMLEITUNG_FALLWEISE** **#GBG** **#GEBUEHREN_ANZEIGER** **#DM_IDENTIFIZIERUNG** **#DM_NOT_SELECTED**	Aktivierte Dienstmerkmale, Octet 3 *Sperre gegen abgehende Verbindung* *Anrufweiterschaltung I* *Anrufweiterschaltung II* *Anschluss gehört zu GBG* *Gebührenanzeiger eingerichtet* *Rufnummeridentifizierung böswilliger Anrufer eingerichtet* *Special code: unselected byte,DM-?? fields containing this pattern will not be coded into the IE*

6.2.6. STATUS DES GERUFENEN TLN (I#SDT)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->SDT_STATUS	Status, Octet 3
#KEINE_ANGABE	*keine Angabe*
#TLN_WIRD_GERUFEN	*Teilnehmer wird angerufen*

6.2.7. ADDITIONALTRANSMISSION ATTRIBUTES (I#ATA)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	1 TR6 Reference
->ATA_FLAG	Flag, Octet 3
#INDICATION	*Indication*
#REQUEST	*Request*
->ATA_ATTRIBUTES	Attributes, Octet 3
#KEIN_SATELLIT	*Keine Satellitenverbindung*
#EIN_SATELLIT	*Eine Satellitenverbindung*
#ZWEI_SATELLIT	*Zwei Satellitenverbindungen*
#DREI_SATELLIT	*Drei Satellitenverbindungen*

ISDN LAYER 3 EUROPEAN VARIANT (ETSI)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2 Rev.1
Revision Date	:	May, 1992
Engineering	:	AUT E33 D / MNO
Quality Control	:	MWB KQP R.

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. Shift IE (I#SHIFT) 6-1

6.2. Congestion Level IE (I#CONG_LEVEL) 6-1

6.3. Segmented Message IE (I#SEG_MSG) 6-2

6.4. Bearer Capability IE (I#BEARER_CAP) 6-2

6.5. Cause IE (I#CAUSE) 6-7

6.6. Call Identity IE (I#CALL_ID) 6-9

6.7. Call State IE (I#CALL_STATE) 6-10

6.8. Channel Identification IE (I#CHANNEL_ID) 6-11

6.9. Facility IE (I#FACIL) 6-12

6.10. Progress Indicator IE (I#PROGRESS_IND) 6-15

6.11. Network-Specific Facility IE (I#NETWORK_FACIL) 6-16

6.12. Notification Indicator IE (I#NOTIFIC_IND) 6-16

6.13. Display IE (I#DISPLAY) 6-17

6.14. Date/Time IE (I#DATE) 6-17

6.15. Keypad IE (I#KEYPAD) 6-17

6.16. Signal IE (I#SIGNAL) 6-18

6.17. Calling Number IE (I#CALLING_NUM) 6-19

6.18. Calling Subaddress IE (I#CALLING_SAD) 6-20

6.19. Called Number IE (I#CALLED_NUM) 6-20

6.20. Called Subaddress IE (I#CALLED_SAD) 6-21

6.21. Redirecting Number IE (I#REDIRECTING_NUM) 6-21

6.22. Transit Network Selection IE (I#TRANS_NW_SEL) 6-22

6.23. Restart Indicator IE (I#RESTART_IND) 6-23

6.24. Low Layer Compatability IE (I#LOW_LAY_COMP) 6-24
6.25. High Layer Compatability IE (I#HI_LAY_COMP) 6-29
6.26. User User Information IE (I#UU_INFO) 6-30
6.27. Escape for Extension IE (I#ESCAPE) 6-30

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, Section 3 lists the **Information Element** (IE) Identifiers, while Section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while Section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC** / ***COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, $10 [access] or the ISDN EMULATION / SIMULATION Manual, $6 [set]).

2. MESSAGE TYPE IDENTIFIER

Message Identifier	T/S 46-30 reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknownledge
M#CON_CON	Congestion Control
M#FAC	Facility
M#INFO	Info
M#NOTIFY	Notify
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry
M#SEGMENT	Segmented Message

3. INFORMATION ELEMENT TYPE IDENTIFIER

IE Identifier	T/S 46-30 reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#TRANS_NW_SEL	Transit Network Selection
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information
I#ESCAPE	Escape for Extension

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,	OCTET_3.1,	OCTET_3.2,
OCTET_3.3,				
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,	OCTET_6.1,	OCTET_6.2,	OCTET_6.3,
OCTET_6.4,	OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T301	#T301	180	180
Timer T302	#T302	15	10
Timer T303	#T303	4	4
Timer T304	#T304	30	20
Timer T305	#T305	30	30
Timer T306	#T306	30	30
Timer T307	#T307	180	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	40	35
Timer T312	#T312	6	6
Timer T313	#T313	4	4
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	60	60
Timer T318	#T318	4	4
Timer T319	#T319	4	4
Timer T320	#T320	30	30
Timer T321	#T321	30	30
Timer T322	#T322	4	4
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. Shift IE (I#SHIFT)

Selector Name and related Constants	T/S 46-30 Reference
->SH_TYPE	Locking/Non-locking Shift Type
#NON_LOCKING	*Non Locking*
#LOCKING	*Locking*
->SH_CODESET	Codeset Specified in Shift IE
#CODESET0	*Q.931 IE*
#CODESET5	*CEPT defined IE*
#CODESET6	*National network specific IE*
#CODESET7	*User Specific IE*

6.2. Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	T/S 46-30 Reference
->CL_CONGESTION_LEVEL	Congestion Level
#RECEIVER_READY	*Receiver Ready*
#RECEIVER_NOT_READY	*Receiver not Ready*

6.3. Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.4. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_7

Selector Name and related Constants	T/S 46-30 Reference
->BC_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for present network*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*

Selector Name and related Constants	T/S 46-30 Reference
->BC_TRANSFER_RATE	Information Transfer Rate, Octet 4
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_STRUCTURE	Structure, Octet 4a
#DEFAULT	*Default*
#8KHZ_INTEGRITY	*8 kHz Integrity*
#SDU_INTEGRITY	*Service Data Unit Integrity*
#UNSTRUCTURED	*Unstructured*
->BC_CONFIGURATION	Configuration, Octet 4a
#POINT_TO_POINT	*Point-to-point*
->BC_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->BC_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
->BC_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->BC_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->BC_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.7XX 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->BC_SYNC/ASYNC	Synchronous or Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*

Selector Name and related Constants	T/S 46-30 Reference
->BC_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->BC_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6 and X.1*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation V.6 and X.1*
#1.2/0.075KBIT/S	*1.2/0.075 kbit/s Recommendation V.6 and X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->BC_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*not used*
#INT_8KBIT/S	*8 kbit/s*
#INT_16KBIT/S	*16 kbit/s*
#INT_32KBIT/S	*32 kbit/s*
->BC_NIC_ON_TX	Network internal clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->BC_NIC_ON_RX	Network internal clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*

Selector Name and related Constants	T/S 46-30 Reference
->BC_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*No Flow Control required*
#REQUIRED	*Flow Control required*
->BC_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept Flow Control*
#ACCEPT	*Can accept Flow Control*
->BC_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->BC_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->BC_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*
->BC_LLI_NEG	Logical Link negotiation, Octet 5b.2
#DEFAULT	*Default, LLI=256 only*
#FULL	*Full protocol negotiation*
->BC_ASSIGNER	Assignor/Assignee, Octet 5b.2
#DEFAULT	*Originator is 'Default assigner'*
#ONLY_ASSIGNER	*Originator is 'Assigner only'*
->BC_IN/OUT_BAND	In/Out-band negotiation, Octet 5b.2
#OUT_BAND	*Negotiation is done 'Out-band'*
#IN_BAND	*Negotiation is done 'In-band'*
->BC_STOP_BITS	Number of stop bits, Octet 5c
#NOT_USED	*Not used*
#1_STOP_BIT	*1 bit*
#1.5_STOP_BITS	*1.5 bits*
#2_STOP_BITS	*2 bits*
->BC_DATA_BITS	Number of data bits, Octet 5c
#NOT_USED	*Not used*
#5_DATA_BITS	*5 bits*
#7_DATA_BITS	*7 bits*
#8_DATA_BITS	*8 bits*

Selector Name and related Constants	T/S 46-30 Reference
->BC_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity information, Octet 5c *odd* *even* *none* *forced to 0* *forced to 1*
->BC_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*
->BC_MODEM_TYPE #V.21 #V.22 #V.22_BIS #V.23 #V.26 #V.26_BIS #V.26_TER #V.27 #V.27_BIS #V.27_TER #V.29 #V.32 #V.35	Modem type, Octet 5d *V.21* *V.22* *V.22 bis* *V.23* *V.26* *V.26 bis* *V.26 ter* *V.27* *V.27 bis* *V.27 ter* *V.29* *V.32* *V.35*
->BC_LAYER2_ID #LAYER2	Layer Identification, Octet 6 *User Info Layer 2 Protocol*
->BC_L2_PROTOCOL #Q.921 #X.25_LINK	Layer 2 Protocol Identifier, Octet 6 *Recommendation Q.921/I.441* *Recommendation X.25, Link Layer*
->BC_LAYER3_ID #LAYER3	Layer 3 Identification, Octet 7 *User Info Layer 3 Protocol*
->BC_L3_PROTOCOL #Q.931 #X.25_PACKET	Layer 3 Protocol Identifier, Octet 7 *Recommendation Q.921/I.441* *Recommendation X.25, Packet Layer*

Note 1 : Octet 5b.1 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->BC_L1_PROTOCOL = #RATE_ADAPTION_V120

6.5. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_5

Selector Name and related Constants	T/S 46-30 Reference
->C_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other international Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->C_LOCATION	Location, Octet 3
#USER	*User*
#LOACAL_PRIVATE	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#TRANSIT	*Transit Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International Network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->C_RECOMMENDATION	Recommendation, Octet 3a
#REC_Q.931	*Recommendation Q.931*
#REC_X.21	*Recommendation X.21*
#REC_X.25	*Recommendation X.25*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL0	*Normal Event Class 0*
->C_CAUSE_VALUE	Cause Value, Octet 4
#UNASSIGNED_NUMBER	*Unassigned Number*
#NO_ROUTE_TO_TRANSIT	*No Route to Transit Network*
#NO_ROUTE_TO_DEST	*No Route to Destination*
#CHANNEL_UNACCEPTABLE	*Channel Unacceptable*
#CALL_AWARDED	*Call awarded*
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NON_SELECTED_CLEAR	*Non selected user clearing*

Selector Name and related Constants	T/S 46-30 Reference
#OUT_OF_SERVICE #INVALID_NUMBER_FORMAT #FACILITY_REJECTED #STATUS_ENQ_RESPONSE #NORMAL_UNSPECIFIED	*Destination out of order* *Invalid Number Format* *Facility Rejected* *Response to STATUS ENQUIRY* *Normal, Unspecified*
->C_CAUSE_CLASS #RESOURCE_UNAVAIL ->C_CAUSE_VALUE #NO_CHANNEL_AVAIL #OUT_OF_ORDER #TEMPORARY_FAILURE #SWITCH_CONGESTION #ACCESS_INFO_DISCARD #CIRCUIT_UNAVAIL #RESOURCES_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Resource not Available* Cause Value, Octet 4 *No Circuit/Channel Available* *Network Out of Order* *Temporary Failure* *Switching Equipment Congestion* *Access Information Discarded* *Request Circuit not Available* *Resources not Available*
->C_CAUSE_CLASS #SERVICE_UNAVAIL ->C_CAUSE_VALUE #QUALITY_UNAVAIL #FAC_NOT_SUBSCRIBED #BEARER_NOT_AUTHOR #BEARER_UNAVAIL #SERVICE_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Service not Available* Cause Value, Octet 4 *Quality of service unavailable* *Requested facility not subscribed* *Bearer Capability not Authorized* *Bearer Capability not Available* *Service not Available*
->C_CAUSE_CLASS #SERVICE_UNIMPL ->C_CAUSE_VALUE #BEARER_SERVICE_UNIMPL #CHANNEL_TYPE_UNIMPL #FAC_UNIMPL #ONLY_RESTRICTED #SERVICE_UNIMPL_UNSPEC	Cause Class, Octet 4 *Service not implemented* Cause Value, Octet 4 *Bearer capability not implemented* *Channel Type not Implemented* *Requested facility not implemented* *Restricted only* *Service not Implemented*
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTENT #CALL_ID_NONEXISTENT #CALL_ID_IN_USE #NO_CALL_SUSPENDED #CALL_CLEARED #NONEXISTENT_CUG #USER_NOT_MEMBER_CUG #USER_BELONGS_CUG	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Call Identity does not Exist* *Call Identity in Use* *No call suspended* *Call has been cleared* *Non-existent closed user group* *Called user not member of CUG* *Called user belongs to CUG*

Selector Name and related Constants	T/S 46-30 Reference
#INV_TRANS_NET	*Invalid transit network selection*
#INVALID_MESSAGE_UNSPEC	*Invalid Message*
->C_CAUSE_CLASS	Cause Class, Octet 4
#PROTOCOL_ERROR	*Protocol Error*
->C_CAUSE_VALUE	Cause Value, Octet 4
#MAND_IE_MISSING	*Mandatory IE is Missing*
#MESSAGE_TYPE_UNIMPL	*Message Type Non-existing*
#MESSAGE_INCOMPAT	*Messgae not Compatible*
#IE_UNIMPL	*IE Non-existing*
#INVALID_IE_CONTENTS	*Invalid IE Contents*
#MESSAGE_UNDEFINED	*Message not compatible with state*
#TIMER_EXPIRY	*Recovery on Timer Expiry*
#PROTOCOL_ERROR_UNSPEC	*Protocol error*
->C_CAUSE_CLASS	Cause Class, Octet 4
#INTERWORKING	*Interworking*
->C_CAUSE_VALUE	Cause Value, Octet 4
#INTERWORK_UNSPEC	*Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.6. Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.7. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->CS_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->CS_CALL_STATE #NULL #CALL_INIT #OVERLAP_SENDING #OUTGOING_CALL_PROC #CALL_DELIVERED #CALL_PRESENT #CALL_RECEIVED #CONNECT_REQUEST #INCOMING_CALL_PROC #ACTIVE #DISC_REQUEST #DISC_INDICATION #SUSPEND_REQUEST #RESUME_REQUEST #RELEASE_REQUEST #CALL_ABORT #OVERLAP_RECEIVING	Call State, Octet 3 *Null* *Call Init* *Overlap Sending* *Outgoing Call Proceeding* *Call Delivered* *Call Present* *Call Request* *Connect Request* *Incomming Call Proceeding* *Active* *Disconnected Request* *Disconnected Indication* *Suspend Request* *Resume Request* *Release Request* *Call Abort* *Overlap Receiving*

6.8. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_3.3

Selector Name and related Constants	T/S 46-30 Reference
->CID_INT_PRESENT #EXPLICIT #IMPLICIT	Interface Identifier Present, Octet 3 *Explicitly Identified* *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Other Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #D_CHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *D-Channel Indentified* *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_INT_ID	Interface identifier, Octet 3.1 *
->CID_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3.2 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 3.2 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS #H0_CHANNEL_UNITS #H11_CHANNEL_UNITS #H12_CHANNEL_UNITS	Channel or Map Type, Octet 3.2 *B-Channel Units* *H0-Channel Units* *H11-Channel Units* *H12-Channel Units*
->CID_NUMBER	Channel number, Octet 3.3

6.9. Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	T/S 46-30 Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	T/S 46-30 Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #USER_NOT_SUBSCRIBED #PROVIDED_BY_NETWORK	Error Value, Octet 6.5 (Note 3) *User not subscribed* *Provided by network*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	T/S 46-30 Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.10. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->PI_CODING_STANDARD	Coding Standard, Octet 3.2
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->PI_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVAT	*Local Private Nertwork*
#LOCAL_PUBLIC	*Local Public Network*
#REMOTE_PUBLIC	*Remote Public Network*
#REMOTE_PRIVATE	*Remote Private Network*
#INTERNAT_NETWORK	*International network*
#BEYOND_INTERWORK	*Network Beyond Interworking Point*
->PI_DESCRIPTION	Progress Description, Octet 4
#NOT_END_TO_END	*Call is not end-to-end ISDN*
#DEST_NON_ISDN	*Destination Address is non-ISDN*
#ORIG_NON_ISDN	*Origination Address is non-ISDN*
#RETURNED_TO_ISDN	*Call has Returned to the ISDN*
#INBAND_INFO_AVAIL	*In-band Info now Available*

6.11. Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3.1, OCTET_3.2, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3.1 *User specified* *National network id.* *International network id.*
->NF_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3.1 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->NF_NET_ID	Network Identification, Octet 3.2 *
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.12. Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE #EXT_TO_COMPONENT	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change* *Extension to component usage*
->NI_COMPONENT	Component, Octet 4 *

6.13. Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->D_DISPLAY	Display, Octet 3 *

6.14. Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7

Selector Name and related Constants	T/S 46-30 Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7

6.15. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->K_KEYPAD	Keypad, Octet 3 *

6.16. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on - Pattern 0*
#ALERTING_ON_1	*Alerting on - Pattern 1*
#ALERTING_ON_2	*Alerting on - Pattern 2*
#ALERTING_ON_3	*Alerting on - Pattern 3*
#ALERTING_ON_4	*Alerting on - Pattern 4*
#ALERTING_ON_5	*Alerting on - Pattern 5*
#ALERTING_ON_6	*Alerting on - Pattern 6*
#ALERTING_ON_7	*Alerting on - Pattern 7*
#ALERTING_OFF	*Alerting off*

6.17. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLGN_NUMBER_TYPE	Type of Number, Octet 3
#UNKNOWN	*Unknown*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
#SUBSCRIBER	*Subscriber Number*
#ABBREVIATED	*Abbreviated Number*
->CLGN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#UNKNOWN_PLAN	*Unknown*
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#DATA_PLAN	*Data Numbering Plan X.121*
#TELEX_PLAN	*Telex Numbering Plan F.69*
#NATIONAL_PLAN	*National Numbering Plan*
#PRIVATE_PLAN	*Private Numbering Plan*
->CLGN_PRESENTATION	Presentation Indicator, Octet 3
#PRESENT_ALLOWED	*Presentation Allowed*
#PRESENT_RESTRICTED	*Presentation Restricted*
#NUMBER_UNAVAIL	*Number Not Available Due to Interworking*
->CLGN_SCREENING	Screening indicator, Octet 3a
#UNSCREENED	*User-Provided, Not Screened*
#VERIFY_PASSED	*User-Provided, Verified and Passed*
#VERIFY_FAILED	*User-Provided, Verified and Failed*
#NETWORK_PROVIDED	*Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.18. Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ODD/EVEN #EVEN_NUMBER #ODD_NUMBER	Odd/even indicator, Octet 3 *Even number of digits* *Odd number of digits*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.19. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLDN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC #SUBSCRIBER #ABBREVIATED	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Standard Defined for Identified Location* *Subscriber Number* *Abbreviated Number*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #DATA_PLAN #TELEX_PLAN #NATIONAL_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Data Numbering Plan X.121* *Telex Numbering Plan F.69* *National Numbering Plan* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.20. Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->CLDS_ADDRESS_TYPE	Type of Subaddress, Octet 3
#USER_SPECIFIED	*User Specified*
#NSAP	*NSAP (X.213/ISO 8348 AD2)*
->CLDS_ODD/EVEN	Odd/even indicator, Octet 3
#EVEN_NUMBER	*Even number of digits*
#ODD_NUMBER	*Odd number of digits*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.21. Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->RDGN_NUMBER_TYPE	Type of Number, Octet 3
#UNKNOWN	*Unknown*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
#SUBSCRIBER	*Subscriber number*
#ABBREVIATED	*Abbreviated Number*
->RDGN_NUMBERING_PLAN	Numbering Plan Identification, Octet 3
#UNKNOWN_PLAN	*Unknown*
#ISDN_PLAN	*ISDN Numbering Plan E.164*
#DATA_PLAN	*Data Numbering Plan X.121*
#TELEX_PLAN	*Telex Numbering Plan F.69*
#NATIONAL_PLAN	*National Numbering Plan*
#PRIVATE_PLAN	*Private Numbering Plan*

Selector Name and related Constants	T/S 46-30 Reference
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->RDGN_REASON #CALL_FORW_BUSY #CALL_FORW_NO_REPLY #CALLED_OUT_OF_ORDER #FORW_BY_CALLED #UNCOND_OR_SYS	Reason for redirection, Octet 3b *Call forwarding or called DTE busy* *Call forwarding no reply* *Called DTE out of order* *Forwarding by called DTE* *Unconditional or systematic red.*
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.22. Transit Network Selection IE (I#TRANS_NW_SEL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->TR_NET_ID_TYPE #USER_SPECIFIED_NET #NATIONAL_NET #INTERNATIONAL_NET	Type of Network Identification, Octet 3 *User specified* *National network id.* *International network id.*
->TR_NET_ID_PLAN #UNKNOWN #CARRIER_CODE #DATA_NET_CODE	Network Identification Plan, Octet 3 *Unknown* *Carrier id. code* *Data id. code (Recommendation X.121)*
->TR_NET_ID	Network Identification, Octet 4 *

6.23. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	T/S 46-30 Reference
->RI_CLASS	Class of Channel or Interface, Octet 3
#INDICATED_CHANNEL	*Indicated Channel*
#SINGLE_INTERFACE	*Single Interface*
#ALL_INTERFACES	*All Interface*

6.24. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	T/S 46-30 Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 2.1 or 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO #VIDEO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio* *Video*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation indicator, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #CHANNEL_SIZE #64KBIT/S #2x64KBIT/S #384KBIT/S #1536KBIT/S #1920KBIT/S	Information Transfer Rate, Octet 4 *Channel size* *64 kbit/s* *2 x 64 kbit/s* *384 kbit/s* *1536 kbit/s* *1920 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY #UNSTRUCTURED	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity* *Unstructured*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*

Selector Name and related Constants	T/S 46-30 Reference
->LL_ESTABLISHMENT	Establishment, Octet 4a
#DEMAND	*Demand*
->LL_SYMMETRY	Symmetry, Octet 4b
#BIDIRECT_SYMMETRIC	*Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B	Information Transfer Rate (D->O), Octet 4b
#CHANNEL_SIZE	*Channel size*
#64KBIT/S	*64 kbit/s*
#2x64KBIT/S	*2 x 64 kbit/s*
#384KBIT/S	*384 kbit/s*
#1536KBIT/S	*1536 kbit/s*
#1920KBIT/S	*1920 kbit/s*
->LL_LAYER1_ID	Layer 1 Identification, Octet 5
#LAYER1	*User Info Layer 1 Protocol*
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ULAW	*Recommendation G.711 u-law*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#G.7XX_VIDEO	*Recommendation G.711 384 kbit/s Video*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*

Selector Name and related Constants	T/S 46-30 Reference
#32KBIT/S	*32kbit/s Recommendation I.460*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*
->LL_INTERIM_RATE	Intermediate Rate, Octet 5b.1 (Note 1)
#NOT_USED	*Not Used*
#INT_8KBIT/S	*8kbit/s*
#INT_16KBIT/S	*16kbit/s*
#INT_32KBIT/S	*32kbit/s*
->LL_NIC_ON_TX	Network Internal Clock on TX, Octet 5b.1
#NOT_REQUIRED	*Data with NIC not required*
#REQUIRED	*Data with NIC required*
->LL_NIC_ON_RX	Network Internal Clock on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot accept data with NIC*
#ACCEPT	*Can accept data with NIC*
->LL_FLOW_CTRL_TX	Flow Control on TX, Octet 5b.1
#NOT_REQUIRED	*Flow Control not required*
#REQUIRED	*Flow Control Required*
->LL_FLOW_CTRL_RX	Flow Control on RX, Octet 5b.1
#NOT_ACCEPT	*Cannot Accept Flow Control*
#ACCEPT	*Can Accept Flow Control*
->LL_RATE_HEADER	Rate Adaption Header, Octet 5b.2 (Note 2)
#NOT_INCLUDED	*Header not included*
#INCLUDED	*Header included*
->LL_MULTI_FRAME	Multi Frame establishment, Octet 5b.2
#NOT_SUPPORTED	*Not supported, only UI allowed*
#SUPPORTED	*Supported*
->LL_MODE	Mode of operation, Octet 5b.2
#BIT_TRANSPARENT	*Bit transparent mode*
#PROTOCOL_SENSITIVE	*Protocol sensitive mode*

Selector Name and related Constants	T/S 46-30 Reference
->LL_LLI_NEG	Logical Link negotiation, Octet 5b.2
#DEFAULT	*Default, LLI=256 only*
#FULL	*Full protocol negotiation*
->LL_ASSIGNER	Assignor/Assignee, Octet 5b.2
#DEFAULT	*Originator is 'Default assigner'*
#ONLY_ASSIGNER	*Originator is 'Assigner only'*
->LL_IN/OUT_BAND	In/Out-band negotiation, Octet 5b.2
#OUT_BAND	*Negotiation is done 'Out-band'*
#IN_BAND	*Negotiation is done 'In-band'*
->LL_STOP_BITS	Number of Stop Bits, Octet 5c
#NOT_USED	*Not used*
#1_STOP_BIT	*1 Bit*
#1.5_STOP_BITS	*1.5 Bits*
#2_STOP_BITS	*2 Bits*
->LL_DATA_BITS	Number of Data Bits, Octet 5c
#5_DATA_BITS	*5 Bits*
#7_DATA_BITS	*7 Bits*
#8_DATA_BITS	*8 Bits*
->LL_PARITY	Parity Information, Octet 5c
#ODD_PARITY	*Odd*
#EVEN_PARITY	*Even*
#NO_PARITY	*None*
#FORCED_TO_0	*Forced to 0*
#FORCED_TO_1	*Forced to 1*
->LL_DUPLEX_MODE	Duplex mode, Octet 5d
#HALF_DUPLEX	*Half duplex*
#FULL_DUPLEX	*Full duplex*
->LL_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
#V.35	*V.35*

Selector Name and related Constants	T/S 46-30 Reference
->LL_LAYER2_ID #LAYER2	Layer 2 Identification, Octet 6 *User Info Layer 2 Protocol*
->LL_L2_PROTOCOL #ISO_1745 #Q.921 #X.25_LINK #X.25_MULTILINK #EXTENDED_LAPB #HDLC_ARM #HDLC_NRM #HDLC_ABM #LAN_LLC #X.75	Layer 2 Protocol Identifier, Octet 6 *Basic mode ISO 1745* *Recommendation Q.921 (I.441)* *Recommendation X.25, Link level* *Recommendation X.25 Multilink* *Extended LAPB; half duplex (T.71)* *HDLC ARM (ISO 4335)* *HDLC NRM (ISO 4335)* *HDLC ABM (ISO 4335)* *LAN Logical link control (ISO 8802/2)* *Rec. X.75, single link procedure*
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID #LAYER3	Layer 3 identification, Octet 7 *User Info Layer 3 Protocol*
->LL_L3_PROTOCOL #Q.931 #X.25_PACKET #ISO_8208 #ISO_8348 #ISO_8473 #T.70	Layer 3 Protocol Identifier, Octet7 *Recommendation Q.931 (I.451)* *Recommendation X.25, Packet Layer* *ISO 8208* *ISO 8348* *ISO 8473* *Rec. T.70, minimum network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30
Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.25. High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_4A

Selector Name and related Constants	T/S 46-30 Reference
->HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#INTERNATIONAL	*Other International Standards*
#NATIONAL	*National Standard*
#NETWORK_SPECIFIC	*Standard Defined for Identified Location*
->HL_INTERPRETATION	Interpretation, Octet 3
#FIRST_CHARACTER	*First HL Characteristics to be Used*
->HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
->HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#MAINTENANCE	*Reserved for Maintenance (Note 1)*
#MANAGEMENT	*Reserved for Management (Note 1)*
->HL_CHARACTER_EXT	Extended High Layer Characteristics, Octet 4a
#TELEPHONY	*Telephony*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Processable from Rec. T.502*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application*
#NOT_AVAIL1	*Not available for assignment*
#NOT_AVAIL2	*Not available for assignment*

Note 1 : Only in this case, Octet 4a may follow

6.26. User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->UU_PROTOCOL_DISCR	Protocol discriminator, Octet 3
#USER_SPECIFIC_PD	*User specific protocol*
#OSI_HIGHER_LAYER	*OSI Higher Layer Protocols*
#X.244	*X.244*
#SYS_MNG_CONVERG	*System mng. convergence function*
#IA5_CHAR	*IA5 characters*
#V.120	*reserved for V.120 rate adaption*
#Q.931_CALL_CONTROL	*Q931 Call Control Messages*
->UU_USER_INFO	User-User Information, Octet 4 *

6.27. Escape for Extension IE (I#ESCAPE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	T/S 46-30 Reference
->EX_IE_ID	Information element identifier, Octet 3
->EX_ESCAPE	Contents of information element, Octet 4 *

ISDN LAYER 3 NATIONAL VARIANT (VN3)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	November, 1992
Engeneering	:	AUT E33 D / Zbk
Quality control	:	MWB KQP

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

3.1. CODESET 0 3-1

3.2. CODESET 6 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. CODESET 0 IEs 6-1
6.1.1 Shift IE (I#SHIFT) 6-1
6.1.2 Congestion Level IE (I#CONG_LEVEL) 6-1
6.1.3 Segmented Message IE (I#SEG_MSG) 6-1
6.1.4 Bearer Capability IE (I#BEARER_CAP) 6-2
6.1.5 Cause IE (I#CAUSE) 6-2
6.1.6 Call Identity IE (I#CALL_ID) 6-5
6.1.7 Call State IE (I#CALL_STATE) 6-5
6.1.8 Channel Identification IE (I#CHANNEL_ID) 6-6
6.1.9 Facility IE (I#FACIL) 6-7
6.1.10 Progress Indicator IE (I#PROGRESS_IND) 6-10
6.1.11 Network-Specific Facility IE (I#NETWORK_FACIL) 6-10
6.1.12 Notification Indicator IE (I#NOTIFIC_IND) 6-10
6.1.13 Display IE (I#DISPLAY) 6-11
6.1.14 Date/Time IE (I#DATE) 6-11
6.1.15 Keypad IE (I#KEYPAD) 6-11
6.1.16 Signal IE (I#SIGNAL) 6-12
6.1.17 Calling Number IE (I#CALLING_NUM) 6-13
6.1.18 Calling Subaddress IE (I#CALLING_SAD) 6-13
6.1.19 Called Number IE (I#CALLED_NUM) 6-14
6.1.20 Called Subaddress IE (I#CALLED_SAD) 6-14
6.1.21 Redirecting Number IE (I#REDIRECTING_NUM) 6-15
6.1.22 Restart Indicator IE (I#RESTART_IND) 6-15
6.1.23 Low Layer Compatability IE (I#LOW_LAY_COMP) 6-16
6.1.24 High Layer Compatability IE (I#HI_LAY_COMP) 6-20
6.1.25 User User Information IE (I#UU_INFO) 6-20

6.2. CODESET 6 IEs 6-21
6.2.1 Terminal Capability IE (I#TERM_CAPAB) 6-21
6.2.2 User User Facility IE (I#UU_FACIL) 6-21

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, section 3 lists the **Information Element** (IE) Identifiers, while section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC / *COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, section10 [access] or the ISDN EMULATION / SIMULATION Manual, section 6 [set]).

2. MESSAGE TYPE IDENTIFIER

Message Identifier	VN3 Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknowledge
M#SEGMENT	Segmented Message
M#CON_CON	Congestion Control
M#FAC	Facility
M#FAC_ACK	Facility Acknowledge
M#FAC_REJ	Facility Reject
M#REG	Register
M#REG_ACK	Register Acknowledge
M#NOTIFY	Notify
M#REG_REJ	Register Reject
M#INFO	Info
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

3. INFORMATION ELEMENT TYPE IDENTIFIER

3.1. CODESET 0

IE Identifier	VN3 Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information

3.2. CODESET 6

IE Identifier	VN3 Reference
I#TERM_CAPAB	Terminal Capabilities
I#UU_FACIL	User-User Facility

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,		
OCTET_3.1,	OCTET_3.2,	OCTET_3.3,		
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,			
OCTET_6.1,	OCTET_6.2,	OCTET_6.3,	OCTET_6.4,	
OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,	
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T302	#T302	–	15
Timer T303	#T303	10	4
Timer T304	#T304	60	–
Timer T305	#T305	4	30
Timer T307	#T307	–	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	60	10
Timer T313	#T313	4	–
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	–	120
Timer T318	#T318	10	–
Timer T319	#T319	4	–
Timer T322	#T322	4	4
Timer T399	#T399	10	–
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. CODESET 0 IEs

6.1.1. Shift IE (I#SHIFT)

Selector Name and related Constants	VN3 Reference
->SH_TYPE #NON_LOCKING #LOCKING	Locking/Non-locking Shift Type *Non Locking* *Locking*
->SH_CODESET #CODESET0 #CODESET6	Codeset Specified in Shift IE *I.451 (Q.931) IE* *Local Network Specific IE*

6.1.2. Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	VN3 Reference
->CL_CONGESTION_LEVEL #RECEIVER_READY #RECEIVER_NOT_READY	Congestion Level *Receiver Ready* *Receiver not Ready*

6.1.3. Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.1.4. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5,

Selector Name and related Constants	VN3 Reference
->BC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *3.1 kHz Audio*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s*
->BC_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->BC_L1_PROTOCOL #G.711_ALAW	Layer 1 Protocol Identifier, Octet 5 *Recommendation G.711 A-law*

6.1.5. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->C_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->C_LOCATION #USER #LOCAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local private network* *Local public network* *Transit network* *Remote public network* *International network* *Network beyond interworking point*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned number*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS #NORMAL1 ->C_CAUSE_VALUE #NORMAL_CLEARING #USER_BUSY #NO_USER_RESPOND #NO_ANSWER #CALL_REJECTED #NUMBER_CHANGED #NUMBER_UNAWARDED #NON_SELECTED_CLEAR #OUT_OF_SERVICE #INVALID_NUMBER_FORMAT #FACILITY_REJECTED #STATUS_ENQ_RESPONSE #NORMAL_UNSPECIFIED	Cause Class, Octet 4 *Normal Event Class 1* Cause Value, Octet 4 *Normal call clearing* *User Busy* *No User Responding* *User alerting, no answer* *Call Rejected* *Number Changed* *Number not awarded* *Non selected user clearing* *Destination out of order* *Invalid Number Format* *Facility rejected* *Response to STATUS ENQUIRY* *Normal, Unspecified*
->C_CAUSE_CLASS #RESOURCE_UNAVAIL ->C_CAUSE_VALUE #NO_CHANNEL_AVAIL #TEMPORARY_FAILURE #SWITCH_CONGESTION #ACCESS_INFO_DISCARD #CIRCUIT_UNAVAIL #RESOURCES_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Resource not Available* Cause Value, Octet 4 *No Circuit/Channel Available* *Temporary Failure* *Switching Equipment Congestion* *Access information discarded* *Request Circuit not Available* *Resources not Available*
->C_CAUSE_CLASS #SERVICE_UNAVAIL ->C_CAUSE_VALUE #QUALITY_UNAVAIL #FAC_NOT_SUBSCRIBED #BEARER_NOT_AUTHOR #BEARER_UNAVAIL #CALL_RESTRICTED #TRANSFER_REJECT #SERVICE_NOT_SUBSCRIBED #SERVICE_UNAVAIL_UNSPEC	Cause Class, Octet 4 *Service not Available* Cause Value, Octet 4 *Quality of service unavailable* *Requested facility not subscribed* *Bearer capability not authorized* *Bearer capability not available* *Call restricted* *Transfer of national call rejected* *Service not subscribed* *Service not Available*
->C_CAUSE_CLASS #SERVICE_UNIMPL ->C_CAUSE_VALUE #BEARER_SERVICE_UNIMPL #CHANNEL_TYPE_UNIMPL #FAC_UNIMPL #SERVICE_UNIMPL_UNSPEC	Cause Class, Octet 4 *Service not implemented* Cause Value, Octet 4 *Bearer capability not implemented* *Channel Type not Implemented* *Requested facility not implemented* *Service not Implemented*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTENT #CALL_ID_NONEXISTENT #CALL_ID_IN_USE #NO_CALL_SUSPENDED #CALL_CLEARED #INCOMPATIBLE_DEST #INVALID_PARAM #INVALID_MESSAGE_UNSPEC	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Call Identity does not Exist* *Call Identity in Use* *No Call suspended* *Call has been cleared* *Incompatible Destination* *Invalid parameter of requested facility* *Invalid Message*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #INVALID_IE_LENGTH #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on timer expiry* *Invalid Length of mandatory IE* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #SERVICE_SUPP_UNAVAIL #TRANSFER_IMPOSSIBLE #PASSAGE_IN_POSITION #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Service support not available* *Transfer of info not possible* *Passage in position of conversation* *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.1.6. Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.1.7. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Received*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*

6.1.8. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->CID_INT_PRESENT #IMPLICIT	Interface Identifier Present, Octet 3 *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Primary rate Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_CODING_STANDARD #CCITT	Coding Standard, Octet 4 *CCITT*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 4 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS	Channel or Map Type, Octet 4 *B-Channel Units*
->CID_NUMBER	Channel number, Octet 5

6.1.9. Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	VN3 Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User-user information*
->CF_ERR_VALUE #USER_NOT_SUBSCRIBED	Error Value, Octet 6.5 (Note 3) *User not subscribed*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.1.10. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->PI_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->PI_LOCATION #USER #LOCAL_PRIVATE #LOCAL_PUBLIC #REMOTE_PUBLIC #REMOTE_PRIVATE #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local private network* *Local public network* *Remote public network* *Remote private network* *International network* *Network beyond interworking point*
->PI_DESCRIPTION #NOT_END_TO_END #DEST_NON_ISDN #ORIG_NON_ISDN #RETURNED_TO_ISDN #INBAND_INFO_AVAIL	Progress Description, Octet 4 *Call is not end-to-end ISDN* *Destination Address is non-ISDN* *Origination Address is non-ISDN* *Call has returned to the ISDN* *In-band info now available*

6.1.11. Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.1.12. Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->NI_DESCRIPTION #USER_SUSPENDED #USER_RESUMED #BEARER_SERVICE_CHANGE	Notification Descriptor, Octet 3 *User suspended* *User resumed* *Bearer service change*

6.1.13. Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->D_DISPLAY	Display, Octet 3 *

6.1.14. Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8

Selector Name and related Constants	VN3 Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7
->DT_SECOND	Second, Octet 8

6.1.15. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.16. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*

6.1.17. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #SUBSCRIBER	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Subscriber Number*
->CLGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED #NUMBER_UNAVAIL	Presentation Indicator, Octet 3 *Presentation allowed* *Presentation restricted* *Number not available due to interworking*
->CLGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.1.18. Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.1.19. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDN_NUMBER_TYPE #BLOCK_NUMBER #OVERLAP_SEND_NUMBER	Type of Number, Octet 3 *Block numbering* *Overlap sending*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.20. Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.1.21. Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	VN3 Reference
->RDGN_NUMBER_TYPE **#INTERNATIONAL** **#NATIONAL**	Type of Number, Octet 3 *Other International Standards* *National Standard*
->RDGN_NUMBERING_PLAN **#ISDN_PLAN**	Numbering Plan Identification, Octet 3 *ISDN Numbering Plan E.164*
->RDGN_PRESENTATION **#PRESENT_ALLOWED** **#PRESENT_RESTRICTED**	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING **#NETWORK_PROVIDED**	Screening indicator, Octet 3a *Network-Provided*
->RDGN_REASON	Reason for redirection, Octet 3b
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.1.22. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->RI_CLASS **#INDICATED_CHANNEL** **#SINGLE_INTERFACE** **#ALL_INTERFACES**	Class of Channel or Interface, Octet 3 *Indicated Channel* *Single Interface* *All Interface*

6.1.23. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	VN3 Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #64KBIT/S #2x64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s* *2 x 64 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC	Symmetry, Octet 4b *Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B #64KBIT/S #2x64KBIT/S	Information Transfer Rate (D->O), Octet 4b *64 kbit/s* *2 x 64 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*

Selector Name and related Constants	VN3 Reference
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#38.4KBIT/S	*38.4kbit/s (not ETSI)*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#38.4KBIT/S_V14	*38.4kbit/s (V14, neither ETSI nor CCITT)*
#57.6KBIT/S_V14	*57.6kbit/s (V14, neither ETSI nor CCITT)*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*

Selector Name and related Constants	VN3 Reference
->LL_INTERIM_RATE #NOT_USED #INT_8KBIT/S #INT_16KBIT/S #INT_32KBIT/S	Intermediate Rate, Octet 5b.1 (Note 1) *Not Used* *8kbit/s* *16kbit/s* *32kbit/s*
->LL_NIC_ON_TX #NOT_REQUIRED #REQUIRED	Network Internal Clock on TX, Octet 5b.1 *Data with NIC not required* *Data with NIC required*
->LL_NIC_ON_RX #NOT_ACCEPT #ACCEPT	Network Internal Clock on RX, Octet 5b.1 *Cannot accept data with NIC* *Can accept data with NIC*
->LL_FLOW_CTRL_TX #NOT_REQUIRED #REQUIRED	Flow Control on TX, Octet 5b.1 *Flow Control not required* *Flow Control Required*
->LL_FLOW_CTRL_RX #NOT_ACCEPT #ACCEPT	Flow Control on RX, Octet 5b.1 *Cannot Accept Flow Control* *Can Accept Flow Control*
->LL_V120_RATE_ADAPT	V.120 rate adaption field, Octet 5b.2 (Note 2)
->LL_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of Stop Bits, Octet 5c *Not used* *1 Bit* *1.5 Bits* *2 Bits*
->LL_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of Data Bits, Octet 5c *Not used* *5 Bits* *7 Bits* *8 Bits*
->LL_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity Information, Octet 5c *Odd* *Even* *None* *Forced to 0* *Forced to 1*
->LL_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*

Selector Name and related Constants	VN3 Reference
->LL_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
->LL_LAYER2_ID	Layer 2 Identification, Octet 6
#LAYER2	*User Info Layer 2 Protocol*
->LL_L2_PROTOCOL	Layer 2 Protocol Identifier, Octet 6
#ISO_1745	*Basic mode ISO 1745*
#Q.921	*Recommendation Q.921 (i.441)*
#X.25_LINK	*Recommendation X.25, Link level*
#X.25_MULTILINK	*Recommendation X.25 Multilink*
#EXTENDED_LAPB	*Extended LAPB; half duplex (T.71)*
#HDLC_ARM	*HDLC ARM (ISO 4335)*
#HDLC_NRM	*HDLC NRM (ISO 4335)*
#HDLC_ABM	*HDLC ABM (ISO 4335)*
#LAN_LLC	*LAN Logical link control (ISO 8802/2)*
#X.75	*Recommendation X.75, single link procedure*
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID	Layer 3 identification, Octet 7
#LAYER3	*User Info Layer 3 Protocol*
->LL_L3_PROTOCOL	Layer 3 Protocol Identifier, Octet7
#Q.931	*Recommendation Q.931 (I.451)*
#X.25_PACKET	*Recommendation X.25, Packet Layer*
#ISO_8208	*ISO 8208*
#ISO_8348	*ISO 8348*
#ISO_8473	*ISO 8473*
#T.70	*Recommendation T.70, min. network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30 or #NON_CCITT

Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.1.24. High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->HL_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
#NATIONAL	*National Standard*
->HL_INTERPRETATION	Interpretation, Octet 3
#NATIONAL_INTERPRET	*National*
#CCITT/CEPT	*CCITT / CEPT*
->HL_PRESENTATION	Presentation Method of Profile, Octet 3
#PROTOCOL_PROFILE	*High Layer Protocol Profile*
->HL_CHARACTER	High Layer Characteristics Id., Octet 4
#TELEPHONY	*Telephony Rec. G.711*
#FAX_GROUP_2/3	*Fax Group 2/3 Rec. T.62*
#FAX_GROUP_4	*Fax Group 4 Rec. T503*
#MIXED_MODE	*Mixed Mode Rec. T.501*
#PROCESSABLE	*Teletex (base mode) Rec. F.220*
#TELETEX	*Teletex Rec. T.62, T.70*
#VIDEOTEX	*Videotex Rec. T.504*
#VIDEOTEX_INTERNAT	*International Videotex F.300, T.101*
#TELEX	*Telex*
#MHS	*Message Handling Systems (MHS)*
#OSI_APPLICATION	*OSI Application Rec. X.200*
#AUDIOVISUAL	*Audiovisual Rec. F.720, H.242*
#UNKNOWN	*Unknown*
#AUDIOGRAPHY	*Audiography*
#AUDIOVIDEOTEX	*Audiovideotex*
#NOT_NORMAL	*Applications not normalized*

6.1.25. User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UU_PROTOCOL_DISCR	Protocol discriminator, Octet 3
#IA5_CHAR	*IA5 characters*
#NT_NT_FACILITY	*Network to Network facility*
#TRANSGROUP	*Transgroup facilities*
->UU_USER_INFO	User-User Information, Octet 4 *

6.2. CODESET 6 IEs

6.2.1. Terminal Capability IE (I#TERM_CAPAB)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->TC_CODING_STANDARD #NATIONAL	Coding Standard, Octet 3 *National Standard*
->TC_DESCRIPTION #NAT_TERMINAL	Capability Description, Octet 3 *National terminal mode*

6.2.2. User User Facility IE (I#UU_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UF_PROTOCOL_DISCR #NT_NT_FACILITY #MINI_MSG #TRANSGROUP	Coding Standard, Octet 3 *Network to Network facility* *Mini message* *Transgroup facilities*
->UF_USER_FACIL	User facility, Octet 4 *

ISDN LAYER 3 NATIONAL VARIANT (VN3)

User Manual for Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	November, 1992
Engeneering	:	AUT E33 D / Zbk
Quality control	:	MWB KQP R.

1. INTRODUCTION 1-1

2. MESSAGE TYPE IDENTIFIER 2-1

3. INFORMATION ELEMENT TYPE IDENTIFIER 3-1

3.1. CODESET 0 3-1

3.2. CODESET 6 3-1

4. OCTET IDENTIFIER 4-1

5. LAYER 3 TIMER 5-1

6. ACCESSING INFORMATION ELEMENT FIELD VALUES 6-1

6.1. CODESET 0 IEs 6-1
6.1.1 Shift IE (I#SHIFT) 6-1
6.1.2 Congestion Level IE (I#CONG_LEVEL) 6-1
6.1.3 Segmented Message IE (I#SEG_MSG) 6-1
6.1.4 Bearer Capability IE (I#BEARER_CAP) 6-2
6.1.5 Cause IE (I#CAUSE) 6-2
6.1.6 Call Identity IE (I#CALL_ID) 6-5
6.1.7 Call State IE (I#CALL_STATE) 6-5
6.1.8 Channel Identification IE (I#CHANNEL_ID) 6-6
6.1.9 Facility IE (I#FACIL) 6-7
6.1.10 Progress Indicator IE (I#PROGRESS_IND) 6-10
6.1.11 Network-Specific Facility IE (I#NETWORK_FACIL) 6-10
6.1.12 Notification Indicator IE (I#NOTIFIC_IND) 6-10
6.1.13 Display IE (I#DISPLAY) 6-11
6.1.14 Date/Time IE (I#DATE) 6-11
6.1.15 Keypad IE (I#KEYPAD) 6-11
6.1.16 Signal IE (I#SIGNAL) 6-12
6.1.17 Calling Number IE (I#CALLING_NUM) 6-13
6.1.18 Calling Subaddress IE (I#CALLING_SAD) 6-13
6.1.19 Called Number IE (I#CALLED_NUM) 6-14
6.1.20 Called Subaddress IE (I#CALLED_SAD) 6-14
6.1.21 Redirecting Number IE (I#REDIRECTING_NUM) 6-15
6.1.22 Restart Indicator IE (I#RESTART_IND) 6-15
6.1.23 Low Layer Compatability IE (I#LOW_LAY_COMP) 6-16
6.1.24 High Layer Compatability IE (I#HI_LAY_COMP) 6-20
6.1.25 User User Information IE (I#UU_INFO) 6-20

6.2. CODESET 6 IEs 6-21
6.2.1 Terminal Capability IE (I#TERM_CAPAB) 6-21
6.2.2 User User Facility IE (I#UU_FACIL) 6-21

1. INTRODUCTION

This guide is intended to provide the information necessary for using the Layer 3 Decoder commands in Test Manager environments. The Layer 3 Software provides a set of predefined constants for Message Type, Information Element and Octet Identifiers, which are used as parameters for the Decoder commands (described in the ISDN TEST MANAGER Manual) as well as for the FORTH programming method of message building (described in the ISDN EMULATION / SIMULATION Manual).

Additionally the Layer 3 Software provides predefined **selector** constants which may be used to access a particular information element field and the possible **field values**. Consequently it is possible to avoid the use of decimal, hexadecimal and binary values when decoding/building Layer 3 messages within Test Manager scenarios. Therefore the test scripts are easier to read and understand.

The following sections list all these predefined constants for the specific variant of Layer 3 Software. Section 2 lists the defined constants that can be used as **Message Type Identifier**, section 3 lists the **Information Element** (IE) Identifiers, while section 4 provides a list of the **Octet Identifiers**.

Section 5 describes the predefined timers, while section 6 provides various **selectors** and **field values** for each octet contained in an Information Element, which can be used in conjunction with the command ***DEC / *COD** to access/set specific fields within an Information Element.
(see the ISDN TEST MANAGER Manual, section10 [access] or the ISDN EMULATION / SIMULATION Manual, section 6 [set]).

2. MESSAGE TYPE IDENTIFIER

Message Identifier	VN3 Reference
M#ALERT	Alerting
M#CALL_PROC	Call Proceeding
M#CONN	Connect
M#CONN_ACK	Connect Acknownledge
M#PROG	Progress
M#SETUP	Setup
M#SETUP_ACK	Setup Acknownledge
M#RES	Resume
M#RES_ACK	Resume Acknownledge
M#RES_REJ	Resume Reject
M#SUSP	Suspend
M#SUSP_ACK	Suspend Acknownledge
M#SUSP_REJ	Suspend Reject
M#USER_INFO	User Information
M#DISC	Disconnected
M#REL	Release
M#REL_COM	Release Complete
M#REST	Restart
M#REST_ACK	Restart Acknowledge
M#SEGMENT	Segmented Message
M#CON_CON	Congestion Control
M#FAC	Facility
M#FAC_ACK	Facility Acknowledge
M#FAC_REJ	Facility Reject
M#REG	Register
M#REG_ACK	Register Acknowledge
M#NOTIFY	Notify
M#REG_REJ	Register Reject
M#INFO	Info
M#STATUS	Status
M#STATUS_ENQ	Status Enquiry

3. INFORMATION ELEMENT TYPE IDENTIFIER

3.1. CODESET 0

IE Identifier	VN3 Reference
I#SHIFT	Codeset Shift
I#MORE_DATA	More Data
I#SEND_COM	Sending Complete
I#CONG_LEVEL	Congestion Level
I#SEG_MSG	Segmented Message
I#BEARER_CAP	Bearer Capability
I#CAUSE	Cause
I#CALL_ID	Call Identity
I#CALL_STATE	Call State
I#CHANNEL_ID	Channel Identification
I#FACIL	Facilities
I#PROGRESS_IND	Progress Indicator
I#NETWORK_FACIL	Network Specific Facilities
I#NOTIFIC_IND	Notification Indicator
I#DISPLAY	Display
I#DATE	Date/Time
I#KEYPAD	Keypad
I#SIGNAL	Signal
I#CALLING_NUM	Calling Party Number
I#CALLING_SAD	Calling Party Subaddress
I#CALLED_NUM	Called Party Number
I#CALLED_SAD	Called Party Subaddress
I#REDIRECTING_NUM	Redirecting Number
I#RESTART_IND	Restart Indicator
I#LOW_LAY_COMP	Low Layer Compatability
I#HI_LAY_COMP	High Layer Compatability
I#UU_INFO	User-User Information

3.2. CODESET 6

IE Identifier	VN3 Reference
I#TERM_CAPAB	Terminal Capabilities
I#UU_FACIL	User-User Facility

4. OCTET IDENTIFIER

The following list of constants has been defined for addressing particular octets within Information Elements:

OCTET_3,	OCTET_3A,	OCTET_3B,		
OCTET_3.1,	OCTET_3.2,	OCTET_3.3,		
OCTET_4,	OCTET_4A,	OCTET_4B,		
OCTET_5,	OCTET_5A,	OCTET_5B,	OCTET_5B.1,	OCTET_5B.2,
OCTET_5C,	OCTET_5D,			
OCTET_6,	OCTET_6A,			
OCTET_6.1,	OCTET_6.2,	OCTET_6.3,	OCTET_6.4,	
OCTET_6.5,	OCTET_6.6,	OCTET_6.7,	OCTET_6.8,	
OCTET_6.8.1,	OCTET_6.8.2,	OCTET_6.8.3,	OCTET_6.8.4,	OCTET_6.8.5,
OCTET_7,	OCTET_7A,			
OCTET_8				

5. LAYER 3 TIMER

TIMER NAME	TIMER CONSTANT	DEFAULT DURATION (in seconds)	
		user side	network side
Timer T302	#T302	–	15
Timer T303	#T303	10	4
Timer T304	#T304	60	–
Timer T305	#T305	4	30
Timer T307	#T307	–	180
Timer T308	#T308	4	4
Timer T309	#T309	90	90
Timer T310	#T310	60	10
Timer T313	#T313	4	–
Timer T314	#T314	4	4
Timer T316	#T316	120	120
Timer T317	#T317	–	120
Timer T318	#T318	10	–
Timer T319	#T319	4	–
Timer T322	#T322	4	4
Timer T399	#T399	10	–
General Purpose Timer 1	#USER1	10	10
General Purpose Timer 2	#USER2	10	10

6. ACCESSING INFORMATION ELEMENT FIELD VALUES

The following sections contain all information for access of the information element parameter list including relevant octet identifier for all information elements, selector names associated with each information element parameter as well as a complete list of predefined constants for each standardized parameter value.

NOTE: Selector names which represent string parameters are indicated with a "*".

6.1. CODESET 0 IEs

6.1.1. Shift IE (I#SHIFT)

Selector Name and related Constants	VN3 Reference
->SH_TYPE #NON_LOCKING #LOCKING	Locking/Non-locking Shift Type *Non Locking* *Locking*
->SH_CODESET #CODESET0 #CODESET6	Codeset Specified in Shift IE *I.451 (Q.931) IE* *Local Network Specific IE*

6.1.2. Congestion Level IE (I#CONG_LEVEL)

Selector Name and related Constants	VN3 Reference
->CL_CONGESTION_LEVEL #RECEIVER_READY #RECEIVER_NOT_READY	Congestion Level *Receiver Ready* *Receiver not Ready*

6.1.3. Segmented Message IE (I#SEG_MSG)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->SM_FIRST_SEGMENT #SUBSEQUENT_SEGMENT #FIRST_SEGMENT	First Segment Indicator, Octet 3 *Subsequent Segment* *First Segment*
->SM_REMAINING_SEGMENTS	Number of remaining segments, Octet 3
->SM_MESSAGE_TYPE	Type of Message, Octet 4

6.1.4. Bearer Capability IE (I#BEARER_CAP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5,

Selector Name and related Constants	VN3 Reference
->BC_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->BC_TRANSFER_CAP #SPEECH #UNRESTRICTED #3.1KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *3.1 kHz Audio*
->BC_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->BC_TRANSFER_RATE #64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s*
->BC_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*
->BC_L1_PROTOCOL #G.711_ALAW	Layer 1 Protocol Identifier, Octet 5 *Recommendation G.711 A-law*

6.1.5. Cause IE (I#CAUSE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->C_CODING_STANDARD #CCITT	Coding Standard, Octet 3 *CCITT*
->C_LOCATION #USER #LOCAL_PRIVATE #LOCAL_PUBLIC #TRANSIT #REMOTE_PUBLIC #INTERNAT_NETWORK #BEYOND_INTERWORK	Location, Octet 3 *User* *Local private network* *Local public network* *Transit network* *Remote public network* *International network* *Network beyond interworking point*
->C_CAUSE_CLASS #NORMAL0 ->C_CAUSE_VALUE #UNASSIGNED_NUMBER	Cause Class, Octet 4 *Normal Event Class 0* Cause Value, Octet 4 *Unassigned number*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS	Cause Class, Octet 4
#NORMAL1	*Normal Event Class 1*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NORMAL_CLEARING	*Normal call clearing*
#USER_BUSY	*User Busy*
#NO_USER_RESPOND	*No User Responding*
#NO_ANSWER	*User alerting, no answer*
#CALL_REJECTED	*Call Rejected*
#NUMBER_CHANGED	*Number Changed*
#NUMBER_UNAWARDED	*Number not awarded*
#NON_SELECTED_CLEAR	*Non selected user clearing*
#OUT_OF_SERVICE	*Destination out of order*
#INVALID_NUMBER_FORMAT	*Invalid Number Format*
#FACILITY_REJECTED	*Facility rejected*
#STATUS_ENQ_RESPONSE	*Response to STATUS ENQUIRY*
#NORMAL_UNSPECIFIED	*Normal, Unspecified*
->C_CAUSE_CLASS	Cause Class, Octet 4
#RESOURCE_UNAVAIL	*Resource not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#NO_CHANNEL_AVAIL	*No Circuit/Channel Available*
#TEMPORARY_FAILURE	*Temporary Failure*
#SWITCH_CONGESTION	*Switching Equipment Congestion*
#ACCESS_INFO_DISCARD	*Access information discarded*
#CIRCUIT_UNAVAIL	*Request Circuit not Available*
#RESOURCES_UNAVAIL_UNSPEC	*Resources not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNAVAIL	*Service not Available*
->C_CAUSE_VALUE	Cause Value, Octet 4
#QUALITY_UNAVAIL	*Quality of service unavailable*
#FAC_NOT_SUBSCRIBED	*Requested facility not subscribed*
#BEARER_NOT_AUTHOR	*Bearer capability not authorized*
#BEARER_UNAVAIL	*Bearer capability not available*
#CALL_RESTRICTED	*Call restricted*
#TRANSFER_REJECT	*Transfer of national call rejected*
#SERVICE_NOT_SUBSCRIBED	*Service not subscribed*
#SERVICE_UNAVAIL_UNSPEC	*Service not Available*
->C_CAUSE_CLASS	Cause Class, Octet 4
#SERVICE_UNIMPL	*Service not implemented*
->C_CAUSE_VALUE	Cause Value, Octet 4
#BEARER_SERVICE_UNIMPL	*Bearer capability not implemented*
#CHANNEL_TYPE_UNIMPL	*Channel Type not Implemented*
#FAC_UNIMPL	*Requested facility not implemented*
#SERVICE_UNIMPL_UNSPEC	*Service not Implemented*

Selector Name and related Constants	VN3 Reference
->C_CAUSE_CLASS #INVALID_MESSAGE ->C_CAUSE_VALUE #INVALID_CALL_REF #CHANNEL_NONEXISTENT #CALL_ID_NONEXISTENT #CALL_ID_IN_USE #NO_CALL_SUSPENDED #CALL_CLEARED #INCOMPATIBLE_DEST #INVALID_PARAM #INVALID_MESSAGE_UNSPEC	Cause Class, Octet 4 *Invalid Message* Cause Value, Octet 4 *Invalid Call Reference Value* *Channel not Existant* *Call Identity does not Exist* *Call Identity in Use* *No Call suspended* *Call has been cleared* *Incompatible Destination* *Invalid parameter of requested facility* *Invalid Message*
->C_CAUSE_CLASS #PROTOCOL_ERROR ->C_CAUSE_VALUE #MAND_IE_MISSING #MESSAGE_TYPE_UNIMPL #MESSAGE_INCOMPAT #IE_UNIMPL #INVALID_IE_CONTENTS #MESSAGE_UNDEFINED #TIMER_EXPIRY #INVALID_IE_LENGTH #PROTOCOL_ERROR_UNSPEC	Cause Class, Octet 4 *Protocol Error* Cause Value, Octet 4 *Mandatory IE is Missing* *Message Type Non-existing* *Messgae not Compatible* *IE Non-existing* *Invalid IE Contents* *Message not compatible with state* *Recovery on timer expiry* *Invalid Length of mandatory IE* *Protocol error*
->C_CAUSE_CLASS #INTERWORKING ->C_CAUSE_VALUE #SERVICE_SUPP_UNAVAIL #TRANSFER_IMPOSSIBLE #PASSAGE_IN_POSITION #INTERWORK_UNSPEC	Cause Class, Octet 4 *Interworking* Cause Value, Octet 4 *Service support not available* *Transfer of info not possible* *Passage in position of conversation* *Interworking*
->C_DIAGNOSTIC	Diagnostic, Octet 5 *

6.1.6. Call Identity IE (I#CALL_ID)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CI_CALL_ID	Call Identity, Octet 3 *

6.1.7. Call State IE (I#CALL_STATE)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->CS_CALL_STATE	Call State, Octet 3
#NULL	*Null*
#CALL_INIT	*Call Init*
#OVERLAP_SENDING	*Overlap Sending*
#OUTGOING_CALL_PROC	*Outgoing Call Proceeding*
#CALL_DELIVERED	*Call Delivered*
#CALL_PRESENT	*Call Present*
#CALL_RECEIVED	*Call Received*
#CONNECT_REQUEST	*Connect Request*
#INCOMING_CALL_PROC	*Incomming Call Proceeding*
#ACTIVE	*Active*
#DISC_REQUEST	*Disconnected Request*
#DISC_INDICATION	*Disconnected Indication*
#SUSPEND_REQUEST	*Suspend Request*
#RESUME_REQUEST	*Resume Request*
#RELEASE_REQUEST	*Release Request*

6.1.8. Channel Identification IE (I#CHANNEL_ID)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5

Selector Name and related Constants	VN3 Reference
->CID_INT_PRESENT #IMPLICIT	Interface Identifier Present, Octet 3 *Implicitly Identified*
->CID_INT_TYPE #OTHER_INTERFACE #BASIC_INTERFACE	Interface Type, Octet 3 *Primary rate Interface* *Basic Interface*
->CID_PREF/EXCL #EXCLUSIVE #PREFERRED	Preferred/Exclusive, Octet 3 *Exclusive* *Preferred*
->CID_DCHANNEL #NOT_D_CHANNEL	D-channel Indicator, Octet 3 *Not D-Channel*
->CID_INFO_CHAN_SEL #NO_CHANNEL #B1_CHANNEL #B2_CHANNEL #ANY_CHANNEL #AS_INDICATED	Information Channel Selection, Octet 3 *No Channel* *B1 Channel* *B2 Channel* *Any Channel* *As indicated in next octet*
->CID_CODING_STANDARD #CCITT	Coding Standard, Octet 4 *CCITT*
->CID_NUMBER/MAP #MAP #NUMBER	Number/Map Type, Octet 4 *Map* *Number*
->CID_CHANNEL/MAP_TYPE #B_CHANNEL_UNITS	Channel or Map Type, Octet 4 *B-Channel Units*
->CID_NUMBER	Channel number, Octet 5

6.1.9. Facility IE (I#FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_6.1, OCTET_6.2, OCTET_6.3, OCTET_6.4, OCTET_6.5, OCTET_6.6, OCTET_6.7, OCTET_6.8, OCTET_6.8.1, OCTET_6.8.2, OCTET_6.8.3, OCTET_6.8.4, OCTET_6.8.5

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_DISCR #SUPPL_SERV	Service Discriminator, Octet 3 *Supplementary service applications*
->CF_CLASS_4 #CONTEXT_SPEC	Class, Octet 4 *Context specific*
->CF_FORM_4 #CONSTRUCT	Form, Octet 4 *Constructor*
->CF_COMP_TAG #INVOKE #RET_RESULT #RET_ERROR #REJECT	Component Tag, Octet 4 *Invoke* *Return result* *Return error* *Reject*
->CF_LENGTH_FORM_5 #ONE_OCTET	Length Format, Octet 5 *Length is one octet*
->CF_COMP_LENGTH	Length of Component, Octet 5
->CF_COMPONENT	Component, Octet 6 (Note 1) *
->CF_CLASS_6 #UNIVERSAL	Class, Octet 6 *Universal*
->CF_FORM_6 #PRIMITIVE	Form, Octet 6 *Primitive*
->CF_INVOKE_ID_TAG	Invoke Identifier Tag, Octet 6
->CF_LENGTH_FORM_6.1 #ONE_OCTET	Length Format, Octet 6.1 *Length is one octet*
->CF_INV_ID_LENGTH	Length of Invoke Identifier, Octet 6.1
->CF_INVOKE_ID	Invoke Identifier, Octet 6.2 *

Selector Name and related Constants	VN3 Reference
->CF_CLASS_6.3 #UNIVERSAL	Class, Octet 6.3 *Universal*
->CF_FORM_6.3 #PRIMITIVE	Form, Octet 6.3 *Primitive*
->CF_OP_VALUE_TAG	Operation Value Tag, Octet 6.3 (Note 2)
->CF_ERR_VALUE_TAG	Error Value Tag, Octet 6.3 (Note 3)
->CF_LENGTH_FORM_6.4 #ONE_OCTET	Length Format, Octet 6.4 *Length is one octet*
->CF_OP_VALUE_LENGTH	Length of Operation Value, Octet 6.4 (Note 2)
->CF_ERR_VALUE_LENGTH	Length of Error Value, Octet 6.4 (Note 3)
->CF_OP_VALUE #USER_USER	Operation Value, Octet 6.5 (Note 2) *User–user information*
->CF_ERR_VALUE #USER_NOT_SUBSCRIBED	Error Value, Octet 6.5 (Note 3) *User not subscribed*
->CF_CLASS_6.6 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.6 *Universal* *Context specific*
->CF_FORM_6.6 #PRIMITIVE #CONSTRUCT	Form, Octet 6.6 *Primitive* *Constructor*
->CF_SEQ_TAG	Sequence Tag, Octet 6.6
->CF_LENGTH_FORM_6.7 #ONE_OCTET	Length Format, Octet 6.7 *Length is one octet*
->CF_SEQ_LENGTH	Length of Sequence, Octet 6.7
->CF_CLASS_6.8 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8 *Universal* *Context specific*
->CF_FORM_6.8 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8 *Primitive* *Constructor*

Selector Name and related Constants	VN3 Reference
->CF_SERVICE_TAG	Service Tag, Octet 6.8
->CF_LENGTH_FORM_6.8.1 #ONE_OCTET	Length Format, Octet 6.8.1 *Length is one octet*
->CF_SERVICE_LENGTH	Length of Service, Octet 6.8.1
->CF_SERVICE #SERVICE1 #SERVICE2 #SERVICE3	Service, Octet 6.8.2 *Service 1* *Service 2* *Service 3*
->CF_CLASS_6.8.3 #UNIVERSAL #CONTEXT_SPEC	Class, Octet 6.8.3 *Universal* *Context specific*
->CF_FORM_6.8.3 #PRIMITIVE #CONSTRUCT	Form, Octet 6.8.3 *Primitive* *Constructor*
->CF_PREF_TAG	Preferred Tag, Octet 6.8.3
->CF_LENGTH_FORM_6.8.4 #ONE_OCTET	Length Format, Octet 6.8.4 *Length is one octet*
->CF_PREF_LENGTH	Length of Preferred, Octet 6.8.4
->CF_PREFERRED #FALSE #TRUE	Form, Octet 6.8.3 *False (service is required)* *True (service is preferred)*

Note 1 : ->CF_COMPONENT is only available if ->CF_COMP_TAG <> #INVOKE or #RET_RESULT or #RET_ERROR

Note 2 : This value is only available if ->CF_COMP_TAG = #INVOKE

Note 3 : This value is only available if ->CF_COMP_TAG = #RET_ERROR

6.1.10. Progress Indicator IE (I#PROGRESS_IND)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->PI_CODING_STANDARD	Coding Standard, Octet 3
#CCITT	*CCITT*
->PI_LOCATION	Location, Octet 3
#USER	*User*
#LOCAL_PRIVATE	*Local private network*
#LOCAL_PUBLIC	*Local public network*
#REMOTE_PUBLIC	*Remote public network*
#REMOTE_PRIVATE	*Remote private network*
#INTERNAT_NETWORK	*International network*
#BEYOND_INTERWORK	*Network beyond interworking point*
->PI_DESCRIPTION	Progress Description, Octet 4
#NOT_END_TO_END	*Call is not end-to-end ISDN*
#DEST_NON_ISDN	*Destination Address is non-ISDN*
#ORIG_NON_ISDN	*Origination Address is non-ISDN*
#RETURNED_TO_ISDN	*Call has returned to the ISDN*
#INBAND_INFO_AVAIL	*In-band info now available*

6.1.11. Network-Specific Facility IE (I#NETWORK_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->NF_NET_ID_LENGTH	Network Identification Length, Octet 3
->NF_FACILITY_CODE	Network Specific Facility Code, Octet 4 *

6.1.12. Notification Indicator IE (I#NOTIFIC_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->NI_DESCRIPTION	Notification Descriptor, Octet 3
#USER_SUSPENDED	*User suspended*
#USER_RESUMED	*User resumed*
#BEARER_SERVICE_CHANGE	*Bearer service change*

6.1.13. Display IE (I#DISPLAY)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->D_DISPLAY	Display, Octet 3 *

6.1.14. Date/Time IE (I#DATE)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4, OCTET_5, OCTET_6, OCTET_7, OCTET_8

Selector Name and related Constants	VN3 Reference
->DT_YEAR	Year, Octet 3
->DT_MONTH	Month, Octet 4
->DT_DAY	Day, Octet 5
->DT_HOUR	Hour, Octet 6
->DT_MINUTE	Minute, Octet 7
->DT_SECOND	Second, Octet 8

6.1.15. Keypad IE (I#KEYPAD)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->K_KEYPAD	Keypad, Octet 3 *

6.1.16. Signal IE (I#SIGNAL)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->SI_VALUE	Signal Value, Octet 3
#DIAL_ON	*Dial Tone on*
#RING_BACK_ON	*Ring Back Tone on*
#INTERCEPT_ON	*Intercept Tone on*
#CONGESTION_ON	*Network Congestion Tone on*
#BUSY_ON	*Busy Tone on*
#CONFIRM_ON	*Confirm Tone on*
#ANSWER_ON	*Answer Tone on*
#CALL_WAITING_ON	*Call Waiting Tone on*
#OFF_HOOK_ON	*Off-hook Warning Tone on*
#TONES_OFF	*Tones off*
#ALERTING_ON_0	*Alerting on – Pattern 0*
#ALERTING_ON_1	*Alerting on – Pattern 1*
#ALERTING_ON_2	*Alerting on – Pattern 2*
#ALERTING_ON_3	*Alerting on – Pattern 3*
#ALERTING_ON_4	*Alerting on – Pattern 4*
#ALERTING_ON_5	*Alerting on – Pattern 5*
#ALERTING_ON_6	*Alerting on – Pattern 6*
#ALERTING_ON_7	*Alerting on – Pattern 7*
#ALERTING_OFF	*Alerting off*

6.1.17. Calling Number IE (I#CALLING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGN_NUMBER_TYPE #UNKNOWN #INTERNATIONAL #NATIONAL #SUBSCRIBER	Type of Number, Octet 3 *Unknown* *Other International Standards* *National Standard* *Subscriber Number*
->CLGN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED #NUMBER_UNAVAIL	Presentation Indicator, Octet 3 *Presentation allowed* *Presentation restricted* *Number not available due to Interworking*
->CLGN_SCREENING #UNSCREENED #VERIFY_PASSED #VERIFY_FAILED #NETWORK_PROVIDED	Screening indicator, Octet 3a *User-Provided, Not Screened* *User-Provided, Verified and Passed* *User-Provided, Verified and Failed* *Network-Provided*
->CLGN_NUMBER	Calling Number Digits, Octet 4 *

6.1.18. Calling Subaddress IE (I#CALLING_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLGS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLGS_ADDRESS	Calling Subaddress Digits, Octet 4 *

6.1.19. Called Number IE (I#CALLED_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDN_NUMBER_TYPE #BLOCK_NUMBER #OVERLAP_SEND_NUMBER	Type of Number, Octet 3 *Block numbering* *Overlap sending*
->CLDN_NUMBERING_PLAN #UNKNOWN_PLAN #ISDN_PLAN #PRIVATE_PLAN	Numbering Plan Identification, Octet 3 *Unknown* *ISDN Numbering Plan E.164* *Private Numbering Plan*
->CLDN_NUMBER	Called Number Digits, Octet 4 *

6.1.20. Called Subaddress IE (I#CALLED_SAD)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->CLDS_ADDRESS_TYPE #USER_SPECIFIED #NSAP	Type of Subaddress, Octet 3 *User Specified* *NSAP (X.213/ISO 8348 AD2)*
->CLDS_ADDRESS	Called Subaddress Digits, Octet 4 *

6.1.21. Redirecting Number IE (I#REDIRECTING_NUM)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_3B, OCTET_4

Selector Name and related Constants	VN3 Reference
->RDGN_NUMBER_TYPE #INTERNATIONAL #NATIONAL	Type of Number, Octet 3 *Other International Standards* *National Standard*
->RDGN_NUMBERING_PLAN #ISDN_PLAN	Numbering Plan Identification, Octet 3 *ISDN Numbering Plan E.164*
->RDGN_PRESENTATION #PRESENT_ALLOWED #PRESENT_RESTRICTED	Presentation Indicator, Octet 3 a *Presentation Allowed* *Presentation Restricted*
->RDGN_SCREENING #NETWORK_PROVIDED	Screening indicator, Octet 3a *Network-Provided*
->RDGN_REASON	Reason for redirection, Octet 3b
->RDGN_NUMBER	Called Number Digits, Octet 4 *

6.1.22. Restart Indicator IE (I#RESTART_IND)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->RI_CLASS #INDICATED_CHANNEL #SINGLE_INTERFACE #ALL_INTERFACES	Class of Channel or Interface, Octet 3 *Indicated Channel* *Single Interface* *All Interface*

6.1.23. Low Layer Compatability IE (I#LOW_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_3A, OCTET_4, OCTET_4A, OCTET_4B, OCTET_5, OCTET_5A, OCTET_5B.1, OCTET_5B.2, OCTET_5C, OCTET_5D, OCTET_6, OCTET_6A, OCTET_7, OCTET_7A

Selector Name and related Constants	VN3 Reference
->LL_CODING_STANDARD #CCITT #INTERNATIONAL #NATIONAL #NETWORK_SPECIFIC	Coding Standard, Octet 3 *CCITT* *Other International Standards* *National Standard* *Standard Defined for Identified Location*
->LL_TRANSFER_CAP #SPEECH #UNRESTRICTED #RESTRICTED #3.1KHZ_AUDIO #7KHZ_AUDIO	Information Transfer Capability, Octet 3 *Speech* *Unrestricted Digital Information* *Restricted Digital Information* *3.1 kHz Audio* *7 kHz Audio*
->LL_NEGOTIATION_IND #NEG_NOT_POSSIBLE #NEG_POSSIBLE	Negotiation, Octet 3a *Out-band negotiation not possible* *Out-band negotiation possible*
->LL_TRANSFER_MODE #CIRCUIT_MODE #PACKET_MODE	Transfer Mode, Octet 4 *Circuit Mode* *Packet Mode*
->LL_TRANSFER_RATE #64KBIT/S #2x64KBIT/S	Information Transfer Rate, Octet 4 *64 kbit/s* *2 x 64 kbit/s*
->LL_STRUCTURE #DEFAULT #8KHZ_INTEGRITY #SDU_INTEGRITY	Structure, Octet 4a *Default* *8 kHz Integrity* *Service Data Unit Integrity*
->LL_CONFIGURATION #POINT_TO_POINT	Configuration, Octet 4a *Point-to-point*
->LL_ESTABLISHMENT #DEMAND	Establishment, Octet 4a *Demand*
->LL_SYMMETRY #BIDIRECT_SYMMETRIC	Symmetry, Octet 4b *Bidirectional Symmetric*
->LL_TRANSFER_RATE_4B #64KBIT/S #2x64KBIT/S	Information Transfer Rate (D->O), Octet 4b *64 kbit/s* *2 x 64 kbit/s*
->LL_LAYER1_ID #LAYER1	Layer 1 Identification, Octet 5 *User Info Layer 1 Protocol*

Selector Name and related Constants	VN3 Reference
->LL_L1_PROTOCOL	Layer 1 Protocol Identifier, Octet 5
#RATE_ADAPTION_V110/X30	*Rate Adaption V.110/X.30*
#G.711_ALAW	*Recommendation G.711 A-law*
#G.721_ADPCM	*Recommendation G.721 32 kbit/s ADPCM*
#G.7XX_AUDIO	*Recommendation G.7XX 7kHz Audio*
#NON_CCITT	*Non-CCITT rate adaption*
#RATE_ADAPTION_V120	*Rate Adaption V.120*
#RATE_ADAPTION_X31	*Rate Adaption X.31 HDLC flag stuffing*
->LL_SYNC/ASYNC	Synchronous/Asynchronous, Octet 5a
#SYNCHRONOUS	*Synchronous*
#ASYNCHRONOUS	*Asynchronous*
->LL_NEGOTIATION	Negotiation, Octet 5a
#NEG_NOT_POSSIBLE	*In-band negotiation not possible*
#NEG_POSSIBLE	*In-band negotiation possible*
->LL_USER_RATE	User rate, Octet 5a
#E_BITS	*Indicated by E-bits, Recommendation I.460*
#0.6KBIT/S	*0.6kbit/s Recommendation V.6 and X.1*
#1.2KBIT/S	*1.2kbit/s Recommendation V.6*
#2.4KBIT/S	*2.4kbit/s Recommendation V.6*
#3.6KBIT/S	*3.6kbit/s Recommendation V.6*
#4.8KBIT/S	*4.8kbit/s Recommendation V.6 and X.1*
#7.2KBIT/S	*7.2kbit/s Recommendation V.6*
#8KBIT/S	*8kbit/s Recommendation I.460*
#9.6KBIT/S	*9.6kbit/s Recommendation V.6 and X.1*
#14.4KBIT/S	*14.4kbit/s Recommendation V.6*
#16KBIT/S	*16kbit/s Recommendation I.460*
#19.2KBIT/S	*19.2kbit/s Recommendation V.6*
#32KBIT/S	*32kbit/s Recommendation I.460*
#38.4KBIT/S	*38.4kbit/s (not ETSI)*
#48KBIT/S	*48kbit/s Recommendation V.6 and X.1*
#56KBIT/S	*56kbit/s Recommendation V.6*
#64KBIT/S	*64kbit/s Recommendation X.1*
#38.4KBIT/S_V14	*38.4kbit/s (V14, neither ETSI nor CCITT)*
#57.6KBIT/S_V14	*57.6kbit/s (V14, neither ETSI nor CCITT)*
#0.100KBIT/S	*0.100 kbit/s Recommendation X.1*
#0.1345KBIT/S	*0.1345 kbit/s Recommendation X.1*
#0.075/1.2KBIT/S	*0.075/1.2 kbit/s Recommendation X.1*
#1.2/0.075KBIT/S	*1.2/0/075 kbit/s Recommendation X.1*
#0.050KBIT/S	*0.050 kbit/s Recommendation V.6 and X.1*
#0.075KBIT/S	*0.075 kbit/s Recommendation V.6 and X.1*
#0.110KBIT/S	*0.110 kbit/s Recommendation V.6 and X.1*
#0.150KBIT/S	*0.150 kbit/s Recommendation V.6 and X.1*
#0.200KBIT/S	*0.200 kbit/s Recommendation V.6 and X.1*
#0.300KBIT/S	*0.300 kbit/s Recommendation V.6 and X.1*
#12KBIT/S	*12 kbit/s Recommendation V.6*

Selector Name and related Constants	VN3 Reference
->LL_INTERIM_RATE #NOT_USED #INT_8KBIT/S #INT_16KBIT/S #INT_32KBIT/S	Intermediate Rate, Octet 5b.1 (Note 1) *Not Used* *8kbit/s* *16kbit/s* *32kbit/s*
->LL_NIC_ON_TX #NOT_REQUIRED #REQUIRED	Network Internal Clock on TX, Octet 5b.1 *Data with NIC not required* *Data with NIC required*
->LL_NIC_ON_RX #NOT_ACCEPT #ACCEPT	Network Internal Clock on RX, Octet 5b.1 *Cannot accept data with NIC* *Can accept data with NIC*
->LL_FLOW_CTRL_TX #NOT_REQUIRED #REQUIRED	Flow Control on TX, Octet 5b.1 *Flow Control not required* *Flow Control Required*
->LL_FLOW_CTRL_RX #NOT_ACCEPT #ACCEPT	Flow Control on RX, Octet 5b.1 *Cannot Accept Flow Control* *Can Accept Flow Control*
->LL_V120_RATE_ADAPT	V.120 rate adaption field, Octet 5b.2 (Note 2)
->LL_STOP_BITS #NOT_USED #1_STOP_BIT #1.5_STOP_BITS #2_STOP_BITS	Number of Stop Bits, Octet 5c *Not used* *1 Bit* *1.5 Bits* *2 Bits*
->LL_DATA_BITS #NOT_USED #5_DATA_BITS #7_DATA_BITS #8_DATA_BITS	Number of Data Bits, Octet 5c *Not used* *5 Bits* *7 Bits* *8 Bits*
->LL_PARITY #ODD_PARITY #EVEN_PARITY #NO_PARITY #FORCED_TO_0 #FORCED_TO_1	Parity Information, Octet 5c *Odd* *Even* *None* *Forced to 0* *Forced to 1*
->LL_DUPLEX_MODE #HALF_DUPLEX #FULL_DUPLEX	Duplex mode, Octet 5d *Half duplex* *Full duplex*

Selector Name and related Constants	VN3 Reference
->LL_MODEM_TYPE	Modem type, Octet 5d
#V.21	*V.21*
#V.22	*V.22*
#V.22_BIS	*V.22 bis*
#V.23	*V.23*
#V.26	*V.26*
#V.26_BIS	*V.26 bis*
#V.26_TER	*V.26 ter*
#V.27	*V.27*
#V.27_BIS	*V.27 bis*
#V.27_TER	*V.27 ter*
#V.29	*V.29*
#V.32	*V.32*
->LL_LAYER2_ID	Layer 2 Identification, Octet 6
#LAYER2	*User Info Layer 2 Protocol*
->LL_L2_PROTOCOL	Layer 2 Protocol Identifier, Octet 6
#ISO_1745	*Basic mode ISO 1745*
#Q.921	*Recommendation Q.921 (I.441)*
#X.25_LINK	*Recommendation X.25, Link level*
#X.25_MULTILINK	*Recommendation X.25 Multilink*
#EXTENDED_LAPB	*Extended LAPB; half duplex (T.71)*
#HDLC_ARM	*HDLC ARM (ISO 4335)*
#HDLC_NRM	*HDLC NRM (ISO 4335)*
#HDLC_ABM	*HDLC ABM (ISO 4335)*
#LAN_LLC	*LAN Logical link control (ISO 8802/2)*
#X.75	*Recommendation X.75, single link procedure*
->LL_L2_PROTOCOL_OPT	Optional layer 2 protocol information, Octet 6a
->LL_LAYER3_ID	Layer 3 identification, Octet 7
#LAYER3	*User Info Layer 3 Protocol*
->LL_L3_PROTOCOL	Layer 3 Protocol Identifier, Octet7
#Q.931	*Recommendation Q.931 (I.451)*
#X.25_PACKET	*Recommendation X.25, Packet Layer*
#ISO_8208	*ISO 8208*
#ISO_8348	*ISO 8348*
#ISO_8473	*ISO 8473*
#T.70	*Recommendation T.70, min. network layer*
->LL_L3_PROTOCOL_OPT	Optional layer 3 protocol information, Octet 7a

Note 1 : Octet 5b.1 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V110/X30 or #NON_CCITT

Note 2 : Octet 5b.2 is only available, if ->LL_L1_PROTOCOL = #RATE_ADAPTION_V120

6.1.24. High Layer Compatability IE (I#HI_LAY_COMP)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->HL_CODING_STANDARD #CCITT #NATIONAL	Coding Standard, Octet 3 *CCITT* *National Standard*
->HL_INTERPRETATION #NATIONAL_INTERPRET #CCITT/CEPT	Interpretation, Octet 3 *National* *CCITT / CEPT*
->HL_PRESENTATION #PROTOCOL_PROFILE	Presentation Method of Profile, Octet 3 *High Layer Protocol Profile*
->HL_CHARACTER #TELEPHONY #FAX_GROUP_2/3 #FAX_GROUP_4 #MIXED_MODE #PROCESSABLE #TELETEX #VIDEOTEX #VIDEOTEX_INTERNAT #TELEX #MHS #OSI_APPLICATION #AUDIOVISUAL #UNKNOWN #AUDIOGRAPHY #AUDIOVIDEOTEX #NOT_NORMAL	High Layer Characteristics Id., Octet 4 *Telephony Rec. G.711* *Fax Group 2/3 Rec. T.62* *Fax Group 4 Rec. T503* *Mixed Mode Rec. T.501* *Teletex (base mode) Rec. F.220* *Teletex Rec. T.62, T.70* *Videotex Rec. T.504* *International Videotex F.300, T.101* *Telex* *Message Handling Systems (MHS)* *OSI Application Rec. X.200* *Audiovisual Rec. F.720, H.242* *Unknown* *Audiography* *Audiovideotex* *Applications not normalized*

6.1.25. User User Information IE (I#UU_INFO)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UU_PROTOCOL_DISCR #IA5_CHAR #NT_NT_FACILITY #TRANSGROUP	Protocol discriminator, Octet 3 *IA5 characters* *Network to Network facility* *Transgroup facilities*
->UU_USER_INFO	User-User Information, Octet 4 *

6.2. CODESET 6 IEs

6.2.1. Terminal Capability IE (I#TERM_CAPAB)

Possible octet inclusions/exclusions:

OCTET_3

Selector Name and related Constants	VN3 Reference
->TC_CODING_STANDARD #NATIONAL	Coding Standard, Octet 3 *National Standard*
->TC_DESCRIPTION #NAT_TERMINAL	Capability Description, Octet 3 *National terminal mode*

6.2.2. User User Facility IE (I#UU_FACIL)

Possible octet inclusions/exclusions:

OCTET_3, OCTET_4

Selector Name and related Constants	VN3 Reference
->UF_PROTOCOL_DISCR #NT_NT_FACILITY #MINI_MSG #TRANSGROUP	Coding Standard, Octet 3 *Network to Network facility* *Mini message* *Transgroup facilities*
->UF_USER_FACIL	User facility, Octet 4 *

X.25 LAYER 3
D-CHANNEL

User Manual for
Protocol Tester K1197

SIEMENS AG

Software Version	:	V3.0
System Version	:	MF 2.2
Revision Date	:	January, 1991

RELEASE NOTES R-1

1. GENERAL 1-1

1.1. INTRODUCTION 1-1

1.2. X.25 PACKET-LAYER 1-1

1.3. INTERFACE BETWEEN LAPD AND PACKET-LAYER 1-1

1.4. TESTMANAGER 1-2

1.5. RESTRICTIONS TO OTHER X.25 TESTPACKAGES RUNNING ON K1197 1-2

2. MENU SYSTEM 2-1

2.1. MOVING AND SELECTING ITEMS 2-1

2.2. MENU ICONS 2-2

2.3. FUNCTION KEYS 2-3

2.4. X.25 REPORT SELECTION 2-4
2.4.1. X.25 PACKET FILTER SELECTION 2-5
2.4.1.1. X.25 CHANNEL SELECTION 2-6
2.4.1.2. X.25 TRIGGER MENU 2-7
2.4.2. X.25 LAYER 3 TRIGGER SELECTION 2-8
2.4.2.1. X.25 REPORT CAPTURE/DISK CALL TRIGGER 2-10
2.4.2.1.1. X.25 LAYER 3 SETUP 2-11
2.4.2.1.2. X.25 LOGICAL LINKS 2-13
2.4.3. X.25 LOGICAL CHANNELS 2-14
2.4.3.1. X.25 LOGICAL CHANNEL SETUP 2-15
2.4.3.1.1. X.25 RECOMMENDATION SETUP 2-16

3. COMMANDS 3-1

3.1. SETUP COMMANDS 3-1
3.1.1. LAYER 3 SETUP COMMANDS 3-1
3.1.1.1. GENERAL SETUP COMMANDS 3-1
3.1.1.2. CHANNEL SPECIFIC SETUP COMMANDS 3-6

3.2. SEND PROTOCOL COMMANDS 3-8
3.2.1. LAYER 3 SEND COMMANDS 3-8

3.3. SEND PACKET COMMANDS 3-10

3.4. MISCELLANEOUS COMMANDS 3-14
3.4.1. PROTOCOL TIMERS 3-14
3.4.2. REFERENCE EMULATION STATE MACHINES 3-14
3.4.2.1. LAYER 3 STATE MACHINES 3-14

4. TEST MANAGER 4-1

4.1. EVENT RECOGNITION WORDS 4-1
4.1.1. ?RX_PACKET 4-1
4.1.2. ?RX 4-1
4.1.3. LCN-MULTIPLEXER 4-2

4.2. LIST OF COMMUNICATION VARIABLES 4-3

Program:	X.25 D-Channel
Software Version:	V3.1
Previous Version:	V3.0
System Version:	MF 2.2.
Revision Date:	January, 1991

☞ *Please note that it is important to read this paper before reading the manual, because some changed functions are described there.*

Enhancements in the User-Interface of the X.25 Application:

USER FACILITIES IN CALL REQUEST PACKET

USER FACILITIES IN CALL CONNECTED PACKET

USER DATA FIELD

REGISTRATION FIELD IN REGISTRATION PACKET

NEGOTIABLE PACKET AND WINDOW SIZE

NEGOTIABLE THROUGHPUT CLASS

SEND/ACCEPT REVERSE CHARGING

DIAGNOSTIC

USER SPECIFIED FACILITY NEGOTIATION

FAST SELECT

FAST SELECT WITH RESTRICTION

HUNT GROUP

CALL REDIRECTION SETUP

HEX EDITOR FOR FACILITIES

DATA PACKET ECHO

FAST SELECT FACILITY MODE

Enhancements of the X.25 Basis:

RESTART, CLEAR and RESET-INDICATION with 4 Byte only will be detected as to short by the DTE.

The length of the User-Data in CALL- and CLEAR-Packets will be computed

The length of the FACILITY-Field will be computed.

CALL USER DATA is enhanced to 128 Byte.

The default value of the PLAYBACK function is now PLAY=ON.

Layer 3 is enhanced to the function RUN_LAYER3.

1. GENERAL

1.1. INTRODUCTION

The X.31 Simulation/Emulation enables the user to send, receive and decode X.25 PDU's on a LAPD-Layer2. Further more, the Q930/Q931-Protocol is running in parallel and should be considered as the Base-Software for this package, because it is added with the X.25 layer3 protocol behaviour. That means, all the basic functions are available from the ISDN-Q931-Software.

The decision between Packet and Signalling data is done by checking the SAPI (values 16 or Ø). There is no special user-interface to support the X.25. All the features for X.25-Packet layer are available by several commands as known from K1197 - X.25 Software.

1.2. X.25 PACKET-LAYER

The packet layer is based on recommendation CCITT-80 or CCITT-84 (switchable). Full emulation of packet layer is supported. By switching off the emulation it is possible to simulate the protocol outside the state-machine. The Implementation contains of several commands (SETUP, send protocol, send packets, receive packets). They are described in the following chapters.

1.3. INTERFACE BETWEEN LAPD AND PACKET-LAYER

The interface between layer 2 and 3 has changed. From view of layer 3 (2) the appropriate service-primitives to layer 2 (3) are used. Moreover, up to 8 layer 2-links are supported. Every layer 2 link can carry up to 4 logical channels (including channel Ø for Restart). Normally it is useful to reserve some layer 2-links as signalling-links and some other as links for packet-data. (Note: Only links with preselected SAPI=16 are recognized as packet data links).

When using Layer3-send-commands on more than one Layer2-Link, it is useful to precede the send commands with the corresponding link selection to be on the right link.

If ISDN L3-Multiplexer is switched on, and a frame containing SAPI=16 is to be multiplexed, the CEI [CES + SAPI] and the LCN [logical channel number] are considered (Refer to ISDN Connection Management).
For packet data the LCN is used for multiplexing instead of the Call-Reference-value for signalling data.
Additionally, the channel structure, which carries the selected channel can be choosen by command, which forces the multiplexer to update the channel structure to the selected one. An example is given in chapter 4.1.3.
The Testscenarios (on the Test-scrip disk) give an example for using both, signalling and packet data, in parallel.

1.4. TESTMANAGER

Some Testmanager-recognition words with the appropride mnemonics for X.25 PDU's were added to the testpack. However the original Testmanager-concept as known from the ISDN-Software, is used.

1.5. RESTRICTIONS TO OTHER X.25 TESTPACKAGES RUNNING ON K1197

The number of logical channels per link is reduced to 4 (including CHØ for Restart).
Packet retransmisson is not supported.

Layer 2-LAPB is completely replaced by Layer 2-LAPD.

2. MENU SYSTEM

The menu system provides an intuitive interface designed to minimize the need to refer to the manual. It includes pop-up help screens to remind the user of meanings of selections available in the menus.

2.1. MOVING AND SELECTING ITEMS

Within a menu, the following keys may be used to move and perform actions:

Cursor up (↑)	Move to next selectable item above this one.
Cursor down (↓)	Move to next selectable item below this one.
Cursor left (←)	Move to previous menu to the left without entering current item. Note: Some menus do not allow this key in order to force the user to select some action through a function key before leaving the menu (e.g: the Hex editor and Message Building System). String and numeric entry fields: Move cursor one character backward.
Cursor right (→)	Move to next menu to the right if one exists. String and numeric entry fields: Move cursor one character forward (first key struck enables editing the already displayed string, which is indicated by a little blinking cursor).
Control cursor up (ctrl-↑)	Move up one page on scrolling menus.
Control cursor down (ctrl-↓)	Move down one page on scrolling menus.
Control shift cursor up (ctrl-shift-↑)	Move to first item in menu.
Control shift cursor down (ctrl-shift-↓)	Move to last item in menu.
Control shift cursor left (ctrl-shift-←)	Close all opened menu windows except the first one.
Control shift cursor right (ctrl-shift-→)	Open all those menu windows which were closed by ctrl-shift-← or a series of cursor left actions.
Shift cursor right (shift-→)	String and numeric entry fields: Shift the characters from this place on one step forward and create an empty place (blank) between the already existing characters.
Shift cursor left (shift-←)	String and numeric entry fields: Delete the character under the cursor and shift the following characters one step backward.
Control cursor right (ctrl-→)	String and numeric entry fields: Move the blinking cursor to the last character in the string.
Control cursor left (ctrl-←)	String and numeric entry fields: Move the blinking cursor to the first character in the string.
Home	Move to first item in menu (same as ctrl-shift-↑).
Return	Select an item, complete entry of a typed value or execute a command.

Back Space (BS), Delete (DEL)	String and numeric entry fields: Erase the character before the cursor.
Space bar	When the cursor is on a field which takes a number of options, the space bar may be used to scroll through the available options. These options may also be accessed by typing cursor right and selecting the desired option by pressing **Return**.
Number and character keys	Search a string from the list of selectable items. Every key struck moves the cursor forward onto the next item to which the string matches. String and numeric entry fields: Erase the old entry and create a new one. If cursor right, ctrl→, shft→ or shft-← was pressed first, edit the already existing string.

2.2. MENU ICONS

The menus use a number of icons to convey information about the options available on the menu. The borders of the menu are of two types: single line and double line. Double line borders are used to indicate the last menu in a chain. There will not be any menus to the right of a double bordered menu.

At the top and bottom of the right border of a menu there will apear arrows if the menu is bigger than what will fit in the window. The arrow indicates that there are more items outside the window in the direction indicated.

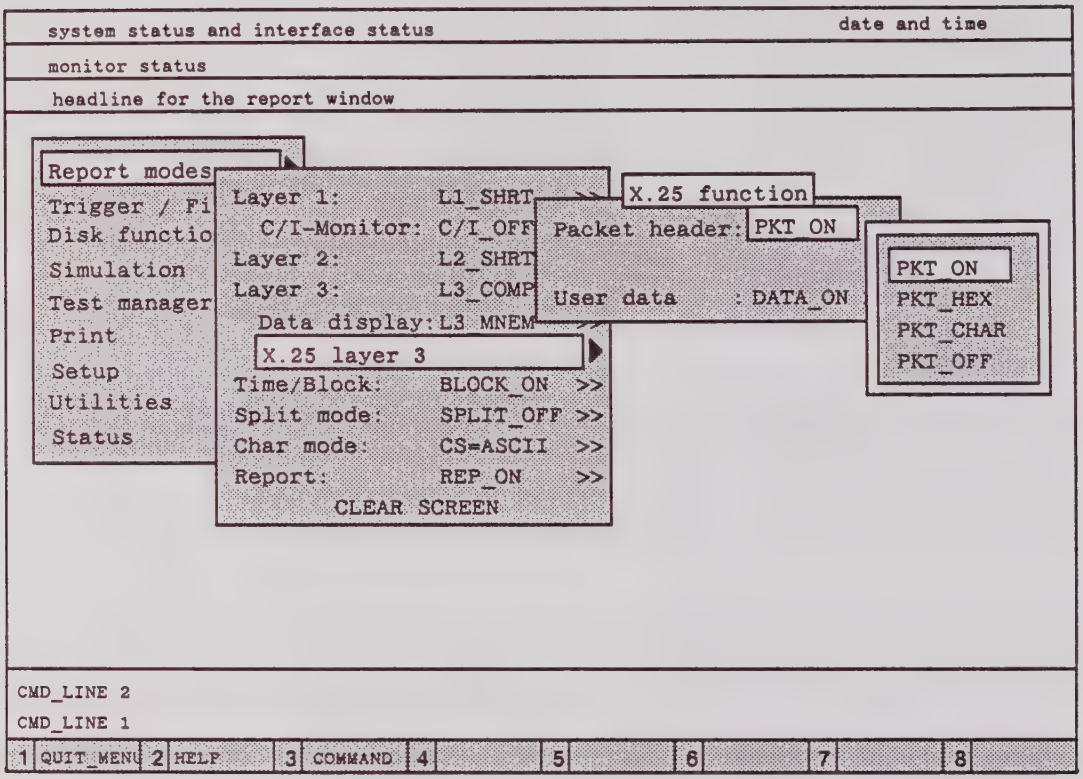
At the right of a selectable item, one of the following may appear: ▶ or ≫. The single solid arrow indicates that there is a menu to the right of this item that will be displayed when cursor right is typed. The double arrow indicates that the field takes a number of options which may be scrolled through with the space bar or listed using the cursor right key.

2.3. FUNCTION KEYS

A set of function keys on the bottom line of the display is almost always availble. Some common keys are listed here.

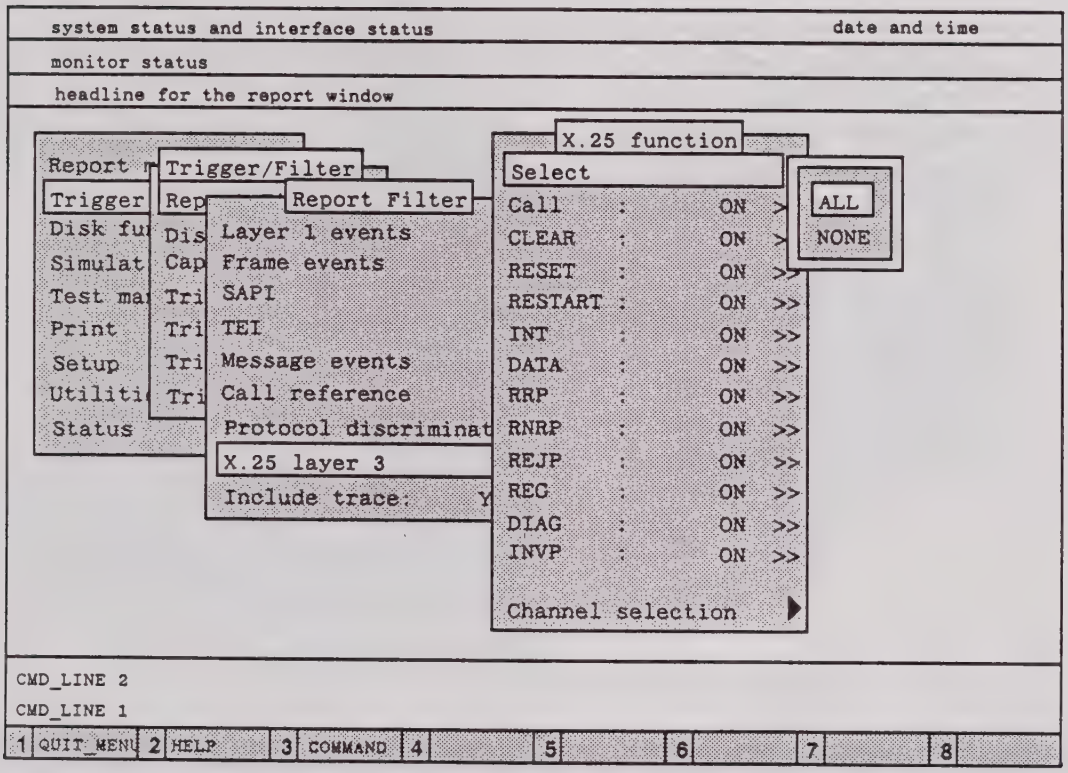
MENU / QUIT_MENU	: Enter to menu system control. / Leaves the menu system control.
HELP / QUIT_HELP	: Open a help menu for the particular menu window.
COMMAND / QUIT_COM	: Give the control to the FORTH interpreter which enables the use of keyboard for typing commands. The menu system window remains still visible. Return the control back to the menu system.
HALT	: Switch to HALT mode
FREEZE	: Switch to FREEZE mode
MONITOR	: Switch to MONITOR mode
TM_KEYS / MON_KEYS	: Switch to Test manager control keys / back to monitor control keys

2.4. X.25 REPORT SELECTION



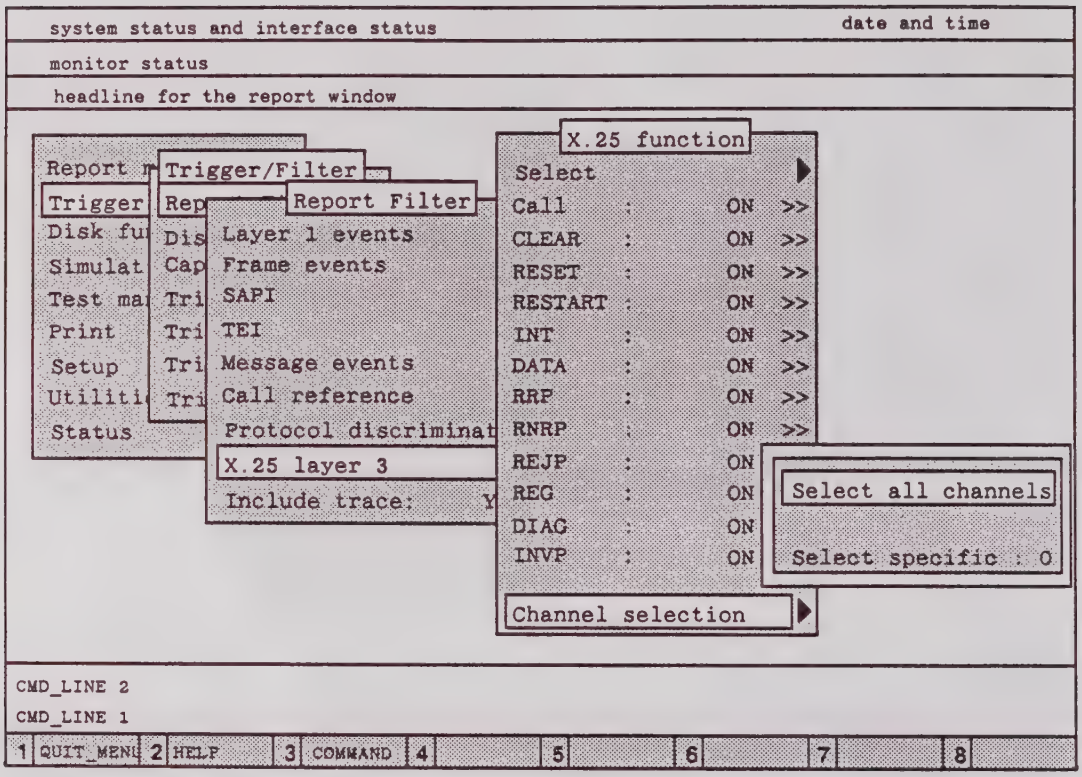
MENU SELECTION:	
PACKET HEADER	: Select complete packet report mode (ON/OFF/ HEX/CHAR)
USER DATA	: Select complete data report mode (ON/OFF/ HEX/CHAR)

2.4.1. X.25 PACKET FILTER SELECTION



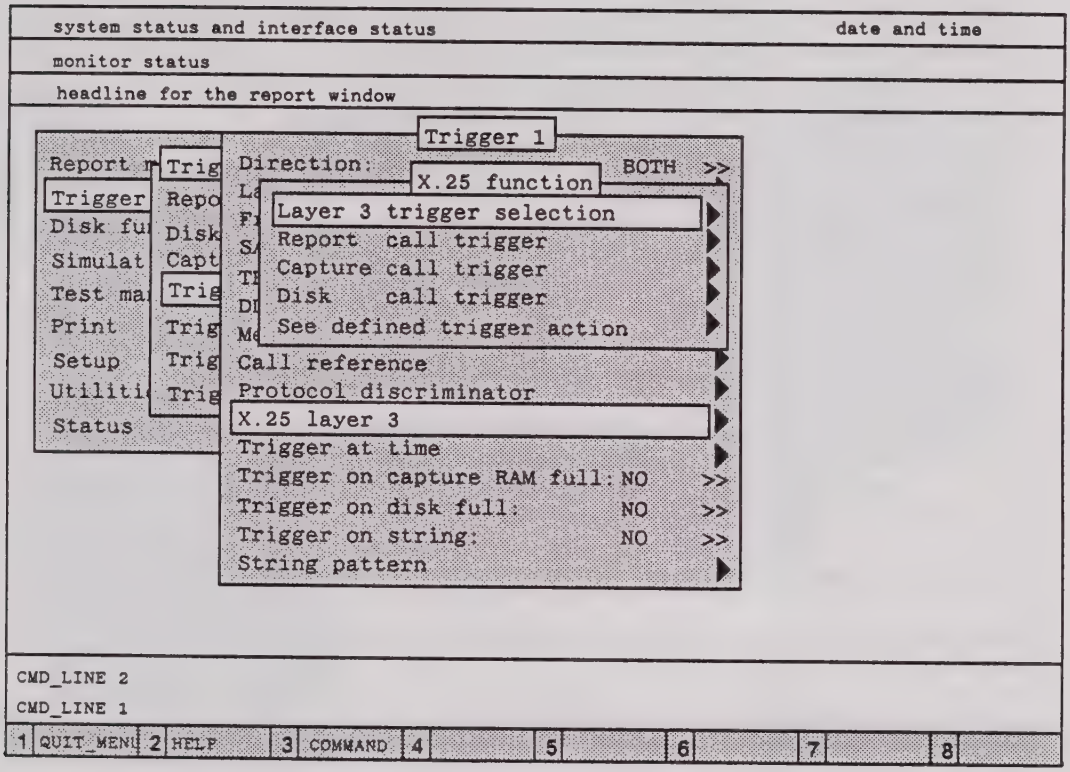
MENU SELECTION:	
ALL	: Enable all layer 3 filter events
NONE	: Disable all layer 3 filter events
CALL	: Enable or disable filter for call
CLEAR	: Enable or disable filter for clear
RESET	: Enable or disable filter for reset
RESTART	: Enable or disable filter for restart
INT	: Enable or disable filter for interrupt
DATA	: Enable or disable filter for data
RRP	: Enable or disable filter for receive ready
RNRP	: Enable or disable filter for receive not ready
REJP	: Enable or disable filter for reject
REG	: Enable or disable filter for registration
DIAG	: Enable or disable filter for diagnostic
INVP	: Enable or disable filter for invalid packet
CHANNEL SELECTION	: Enter channel selection menu

2.4.1.1. X.25 CHANNEL SELECTION



MENU SELECTION:	
SELECT ALL CHANNELS	: Select al logical channels, wanted for report
SELECT SPECIFIC	: Enter specific logical channel, wanted for report

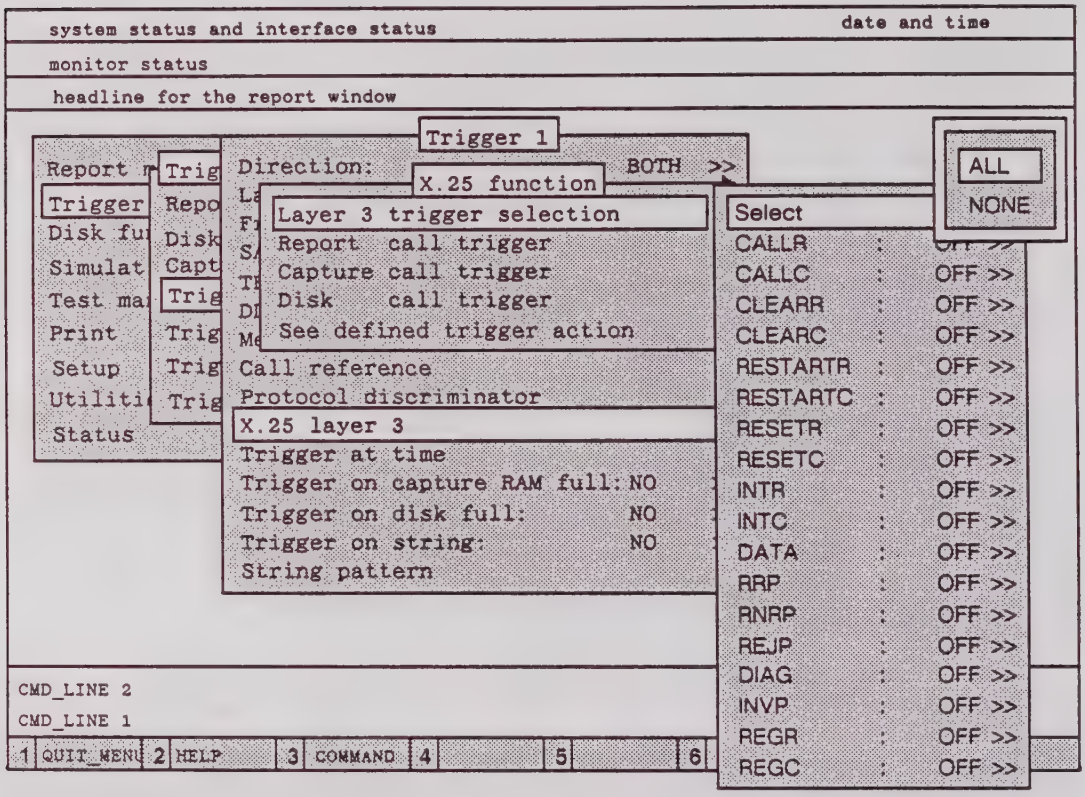
2.4.1.2. X.25 TRIGGER MENU



MENU SELECTION:

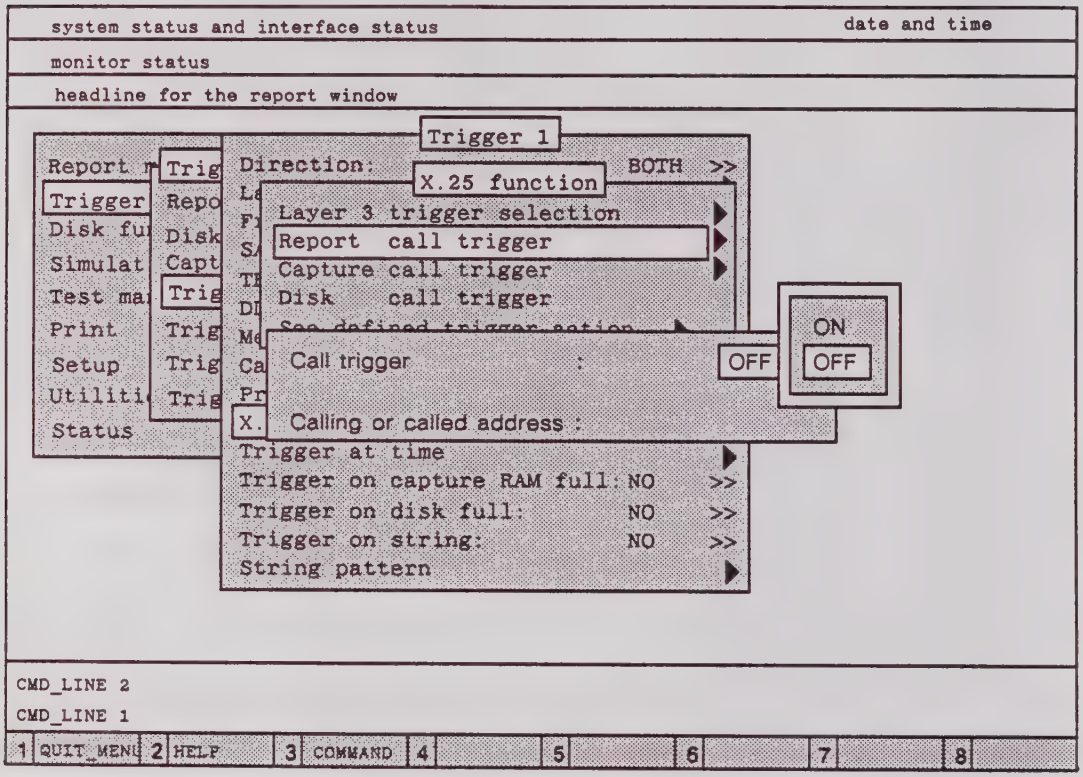
LAYER 3 TRIGGER SELECTION	: Enter menu to select trigger on layer 3 events
REPORT CALL TRIGGER	: Enter report call trigger selection menu
CAPTURE CALL TRIGGER	: Enter capture call trigger selection menu
DISK CALL TRIGGER	: Enter disk call trigger selection menu
SEE DEFINED TRIGGER ACTION	: Enter menu to see current trigger action

2.4.2. X.25 LAYER 3 TRIGGER SELECTION



MENU SELECTION:	
ALL	: Enable all layer 3 trigger events
NONE	: Disable all layer 3 trigger events
CALLR	: Enable or disable trigger for call request
CALLC	: Enable or disable trigger for call counected
CLEARR	: Enable or disable trigger for clear request
CLEARC	: Enable or disable trigger for clear confirmation
RESTARTR	: Enable or disable trigger for restart request
RESTARTC	: Enable or disable trigger for restart confirmation
RESETR	: Enable or disable trigger for reset request
RESETC	: Enable or disable trigger for reset confirmation
INTR	: Enable or disable trigger for interrupt
INTC	: Enable or disable trigger for interrupt confirmation
DATA	: Enable or disable trigger for data
RRP	: Enable or disable trigger for receive ready packet
RNRP	: Enable or disable trigger for receive not ready packet
REJP	: Enable or disable trigger for reject packet
DIAG	: Enable or disable trigger for diagnostic packet
INVP	: Enable or disable trigger for invalid packet
REGR	: Enable or disable trigger for registration request packet
REGC	: Enable or disable trigger for registration confirm packet

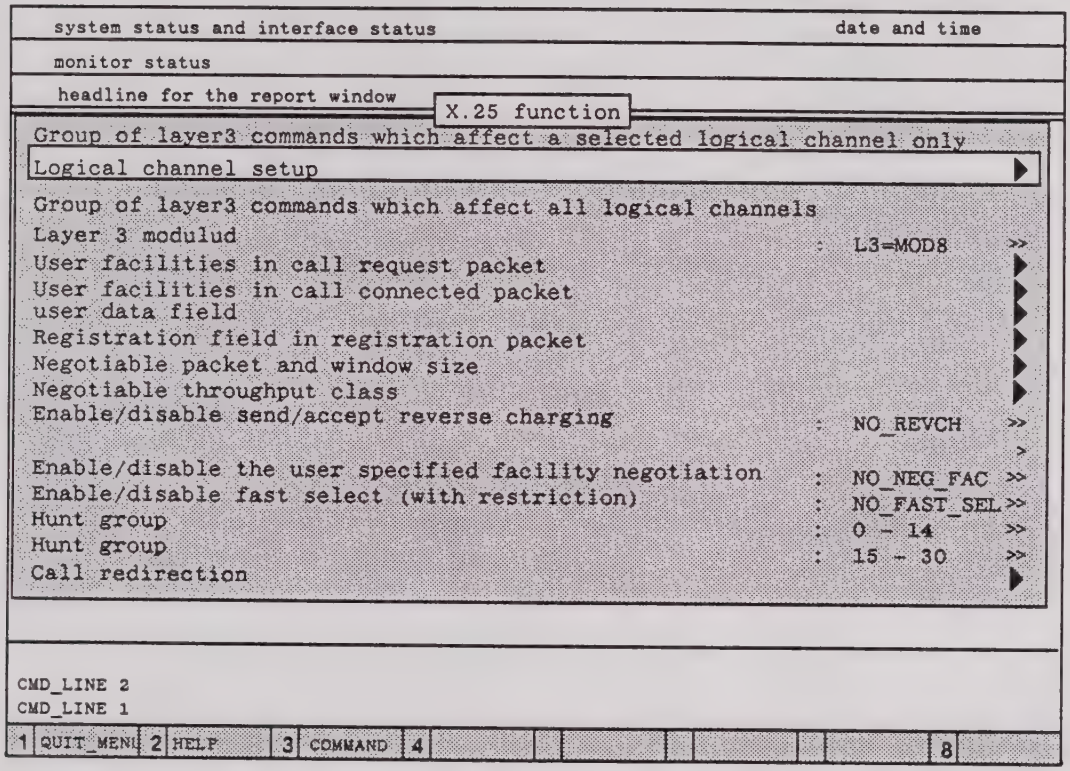
2.4.2.1. X.25 REPORT CAPTURE/DISK CALL TRIGGER



MENU SELECTION:

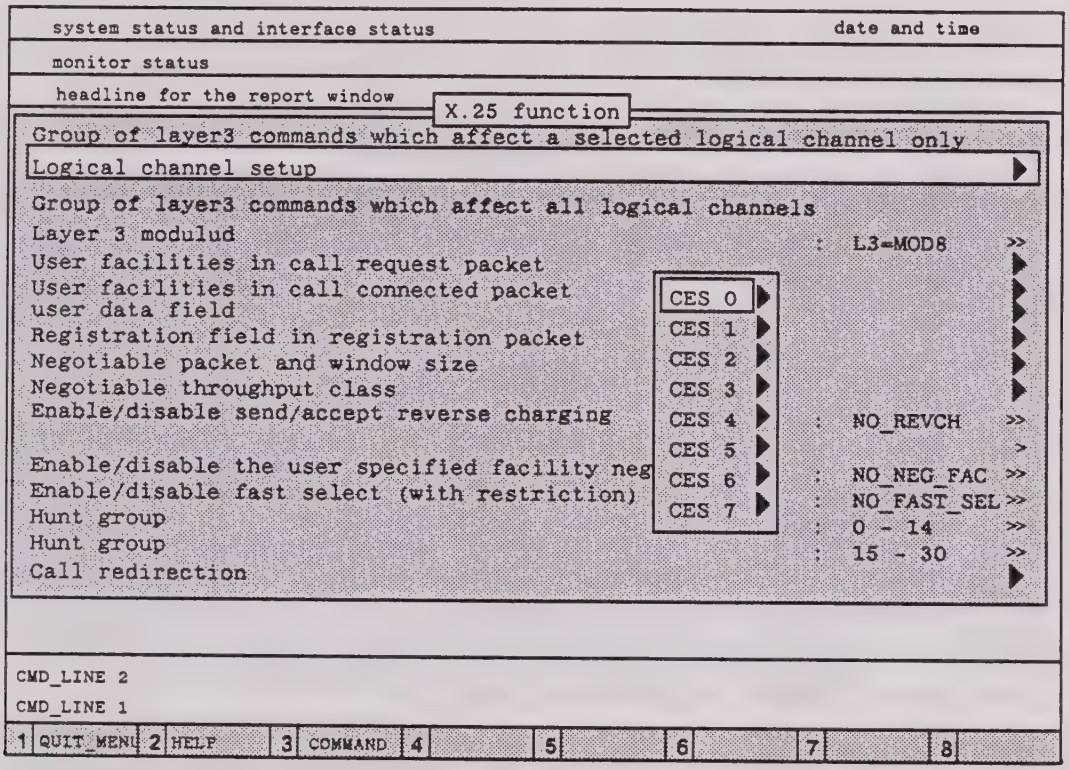
CALL TRIGGER	: Trigger on called/calling address ON or OFF
CALLING OR CALLED ADDRESS	: Enter calling or called address

2.4.2.1.1. X.25 LAYER 3 SETUP


system status and interface status
date and time
monitor status
headline for the report window
X.25 function
Group of layer3 commands which affect a selected logical channel only
Logical channel setup
Group of layer3 commands which affect all logical channels
Layer 3 modulud : L3=MOD8 >>
User facilities in call request packet
User facilities in call connected packet
user data field
Registration field in registration packet
Negotiable packet and window size
Negotiable throughput class
Enable/disable send/accept reverse charging : NO_REVCH >>
Enable/disable the user specified facility negotiation : NO_NEG_FAC >>
Enable/disable fast select (with restriction) : NO_FAST_SEL >>
Hunt group : 0 - 14 >>
Hunt group : 15 - 30 >>
Call redirection
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 8

MENU SELECTION:	
Logical channel setup	: Enter logical link menu
Layer 3 modulus	: Select layer 3 modulus 8 or 128 mode
User facilities in call request packt	: Enter Menu User Facilities in Call Request Packet
User facilities in call connected packt	: Enter Menu User Facilities in Call Connected Packet
User data field	: Enter Menu User Data Field
Negotiable packet and window size	: Enter Menu Negotiable Packet and Window Size
Negotiable throuput class	: Enter Menu Negotiable Throughput Class
Send/access reverse charging	: Enable or disable Send/Access Reverse Charging
Diagnostic packets	: Enable or disable Diagnostic Packets
User specified facility negotiation	: Enable or disable User Specified Facility Negotiation
Fast select (with restriction)	: Enable or disable Fast Select
Hunt group	: Enter Menu Hunt Group
Call redirection setup	: Enter Menu Call Redirection

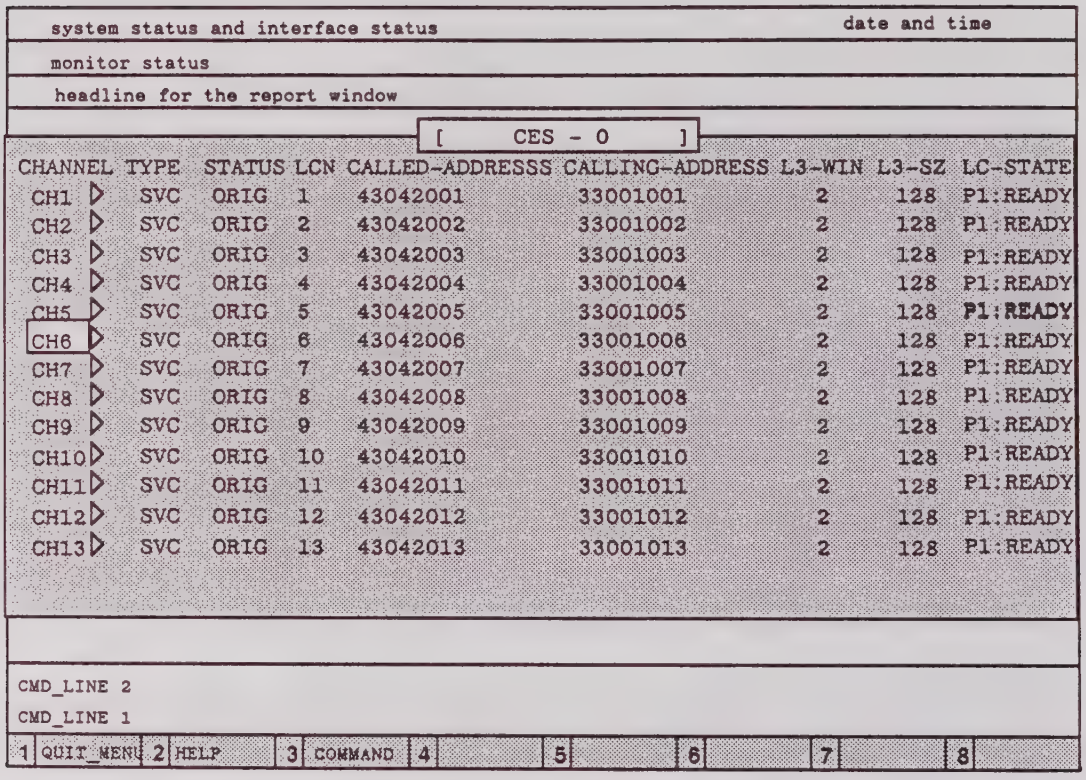
2.4.2.1.2. X.25 LOGICAL LINKS


system status and interface status
date and time
monitor status
headline for the report window
X.25 function
Group of layer3 commands which affect a selected logical channel only
Logical channel setup
Group of layer3 commands which affect all logical channels
Layer 3 modulud : L3=MOD8 >>
User facilities in call request packet
User facilities in call connected packet
user data field
Registration field in registration packet
Negotiable packet and window size
Negotiable throughput class
Enable/disable send/accept reverse charging : NO_REVCH >>
Enable/disable the user specified facility neg : NO_NEG_FAC >>
Enable/disable fast select (with restriction) : NO_FAST_SEL >>
Hunt group : 0 - 14 >>
Hunt group : 15 - 30 >>
Call redirection
CES 0
CES 1
CES 2
CES 3
CES 4
CES 5
CES 6
CES 7
CMD_LINE 2
CMD_LINE 1
1 QUIT MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:

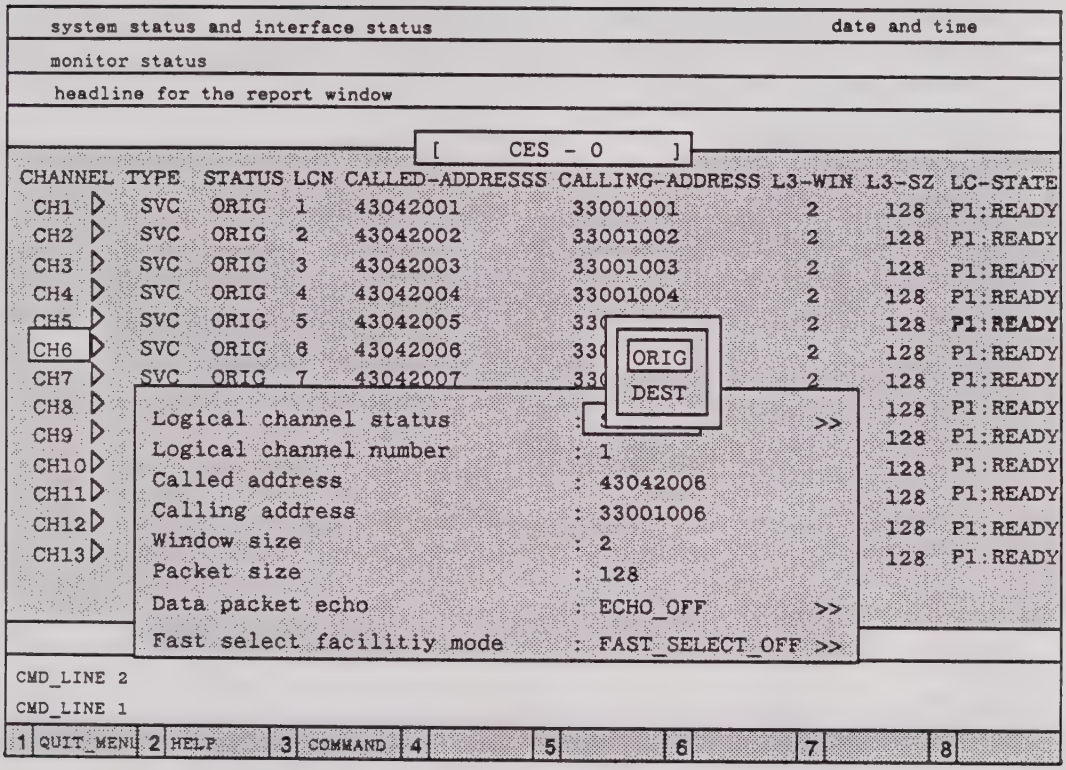
CES0/.../CES7 : Enter logical channel tabe for logical link (0...7)

2.4.3. X.25 LOGICAL CHANNELS


system status and interface status
date and time
monitor status
headline for the report window
[CES - 0]
CHANNEL TYPE STATUS LCN CALLED-ADDRESSS CALLING-ADDRESS L3-WIN L3-SZ LC-STATE
CH1 SVC ORIG 1 43042001 33001001 2 128 P1:READY
CH2 SVC ORIG 2 43042002 33001002 2 128 P1:READY
CH3 SVC ORIG 3 43042003 33001003 2 128 P1:READY
CH4 SVC ORIG 4 43042004 33001004 2 128 P1:READY
CH5 SVC ORIG 5 43042005 33001005 2 128 P1:READY
CH6 SVC ORIG 6 43042006 33001006 2 128 P1:READY
CH7 SVC ORIG 7 43042007 33001007 2 128 P1:READY
CH8 SVC ORIG 8 43042008 33001008 2 128 P1:READY
CH9 SVC ORIG 9 43042009 33001009 2 128 P1:READY
CH10 SVC ORIG 10 43042010 33001010 2 128 P1:READY
CH11 SVC ORIG 11 43042011 33001011 2 128 P1:READY
CH12 SVC ORIG 12 43042012 33001012 2 128 P1:READY
CH13 SVC ORIG 13 43042013 33001013 2 128 P1:READY
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:
CH1/.../CH3 : Enter menu logical channel setup

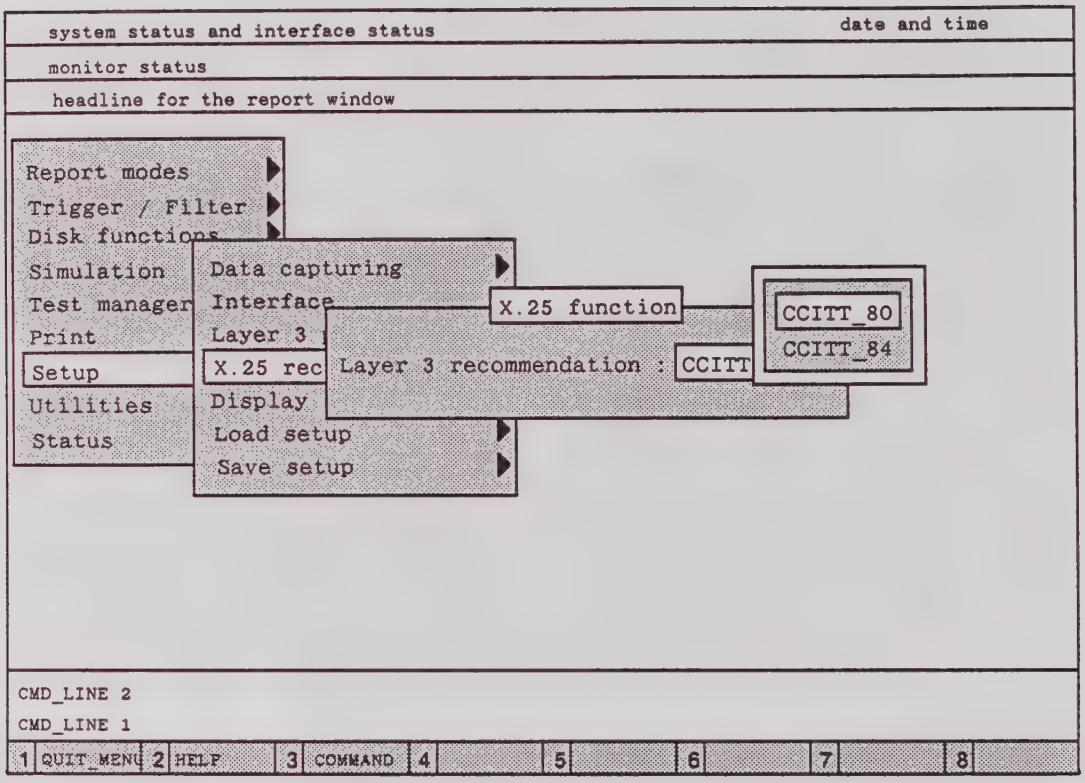
2.4.3.1. X.25 LOGICAL CHANNEL SETUP


system status and interface status
date and time
monitor status
headline for the report window
[CES - 0]
CHANNEL TYPE STATUS LCN CALLED-ADDRESSS CALLING-ADDRESS L3-WIN L3-SZ LC-STATE
CH1 SVC ORIG 1 43042001 33001001 2 128 P1:READY
CH2 SVC ORIG 2 43042002 33001002 2 128 P1:READY
CH3 SVC ORIG 3 43042003 33001003 2 128 P1:READY
CH4 SVC ORIG 4 43042004 33001004 2 128 P1:READY
CH5 SVC ORIG 5 43042005 2 128 P1:READY
CH6 SVC ORIG 6 43042006 2 128 P1:READY
CH7 SVC ORIG 7 43042007 2 128 P1:READY
CH8 128 P1:READY
CH9 128 P1:READY
CH10 128 P1:READY
CH11 128 P1:READY
CH12 128 P1:READY
CH13 128 P1:READY
ORIG
DEST
Logical channel status : >>
Logical channel number : 1
Called address : 43042006
Calling address : 33001006
Window size : 2
Packet size : 128
Data packet echo : ECHO_OFF >>
Fast select facilitiy mode : FAST_SELECT_OFF >>
CMD_LINE 2
CMD_LINE 1
1 QUIT_MENU 2 HELP 3 COMMAND 4 5 6 7 8

MENU SELECTION:

Logical channel status	: Select originate or destination call
Logical channel number	: Enter logical channel value (valid 1 to 4095)
Called address	: Enter called address
Calling address	: Enter calling address
Window size	: Enter packet window size per logical channel
Packet size	: Enter data packet size per logical channel
Data packet echo	: Enable or disable Data Packet Echo
Fast select facility mode	: Select Fast Select Facility Mode

2.4.3.1.1. X.25 RECOMMENDATION SETUP



MENU SELECTION:

Layer 3 recommendation : Select recommendation 80 or 84

3. COMMANDS

This section describes the commands that are used to setup the emulation and to control the protocol operations. These commands are classified into the following categories: SETUP commands, SEND PROTOCOL commands, SEND PACKET commands and Miscellaneous commands. These commands can be either entered directly to the command line or used in the Test Manager scenario contexts.

3.1. SETUP COMMANDS

This section lists the commands for setting up the various user options associated with the protocol.

Layer 3 setup commands concern with the modification of variables and system parameters in Layer 3, and the selection of network utilities in the logical channel. By using this group of commands, the user can change the modulo and the values of the timers used in Layer 3. Furthermore, the user can disable or enable the use of network utility field and/or the user facility field. The user can set the various network utility options, such as the throughput class, the window size, and the packet size. The user can also create his own network utility or user facility field to be used in the virtual call.

3.1.1. LAYER 3 SETUP COMMANDS

The Layer 3 setup commands are divided into two groups. The first group of commands, when issued, will affect all logical channels. The second group of commands will have effect only on the selected logical channels.

3.1.1.1. GENERAL SETUP COMMANDS

CLASS (--)
Set the throughput class facility. The valid values are 75, 150, 300, 600, 1200, 2400, 4800, 9600, 19200 and 48000 (bit/s) according to the Rec. X.25. The default value is set to 9600.
Usage: CLASS 4800

NO_FAC (--)
No user facility field will appear in the Call Request packet. This is the default mode. The only mode for using the user facility field in Call Request packet is by the command USER_FAC which will put the user defined facility field in the Call Request packet.

YES_FAC (--)
Facility negotiation will be used when Call Request Packets are sent. Data packet size, packet window, throughput class and fast select are negotiated. (Automatic default facility)

USER_FAC (--)
This command instructs the emulation to include the user facility field defined by the user with the MAKE_FAC command in the Call Request packet. This command remains in effect until a NO_FAC is issued.

CCITT-80 (--)
This command forces the automatic emulation to satify the Recommendation Version 1980 [3] (Yellow Book).

CCITT-84 (--)
This command forces the automatic emulation to satify the Recommendation Version 1984 [4] (Red Book).

MAKE_FAC **(string --)**
Define the facility field to be used in Call Requests when USER_FAC is selected. The input is in HEX and a space is needed between X" and the input. The first byte in the input is the user facility length with the following bytes to be used in the user facility field. The input should be limited to 64 bytes by CCITT 80 or 109 bytes by CCITT 84 (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional cases. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the facility field in the same order as they are entered.

Example: `X" 03420808" MAKE_FAC`

This will define user facility for packet size of 256 bytes.

NO_CA **(--)**
No address field in the Call Connected packet.

YES_CA **(--)**
The address field received in the Call Request packet will be used in the Call Connected packet. This is the default mode.

NO_CAFAC **(--)**
No user facility field in the Call Connected packet. There are two modes to use the user facility field in the Call Connected Packets:

(i) **ECHO_CAFAC** will echo the user facility field received in the Call Request packet. This is the default mode.

(ii) **USER_CAFAC** will have user defined user facility fields appear in Call Connected packets.

This command will remain in effect until ECHO_CAFAC or USER_CAFAC is issued.

ECHO_CAFAC **(--)**
Echo the facility field received in Call Request packet in the Call Connected packet. This is the default setting.

USER_CAFAC **(--)**
Instruct the emulation to include the user facility field defined by the user using the MAKE_CAFAC command in the Call Connected or Call Request packet. This command remains in effect until NO_CAFAC or ECHO_CAFAC is issued.

MAKE_CAFAC **(string --)**
Define the user facility field to be used in Call Connected packet when USER_CAFAC is selected. The first byte in the input is the user facility length with the following bytes to be used in the user facility field. The input should be limited to 64 bytes (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional conditions. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the facility field in the same order as they are entered.

Example: `X" 03420808" MAKE_CAFAC`

This will define user facility for packet size of 256 bytes.

MAKE_CUD **(string --)**

Make user data field for the next Call Request, Call Connected or Clear Request packet to be sent. There are two usages:

(i) ASCII data **Usage:** `" text input" MAKE_CUD`

(ii) HEX data **Usage:** `X" hex inpuut" MAKE_CUD`

The data field may be up to 16 or 128 octets depending if the Fast Select optional user facility is used. It is the user responsibility to make sure this restriction is being satisfied. The user is allowed to construct a user data field of up to 240 bytes to generate exceptional cases. Inputs that are longer than 240 bytes will be truncated with only the first 240 bytes being used. The input will appear in the user data field in the same order as they are entered, i.e. the first entry will go into the first octet in the user data field. This command remains in effect until 0 **MAKE_CUD** is issued.

Example: `" SIEMENS K1195" MAKE_CUD`

Note: The user data field for Call Connected or Clear Request packet is sent only if the Fast Select network utility option has been selected.

NO_REG **(--)**

No registration field will appear in the Registration packets. This is the default mode. The only mode for using the registration field in Registration packets is by the command USER_REG which will put the user defined registration field in the Registration packets. This is the default mode.

USER_REG **(--)**

This command instructs the emulation to include the registration field defined by the user with the MAKE_REG command in the Registration packets. This command remains in effect until
a NO_REG is issued.

MAKE_REG **(string--)**

Define the registration field to be used in a Registration packet when USER_REG is selected. The input is in HEX and a space is needed between X" and the input. The first byte in the input is the registration length with the following bytes to be used in the registration field. The input should be limited to 64 bytes (including the length) according to the protocol. However, the user can enter length up to 80 bytes to generate exceptional cases. For input that is longer than 80, only the first 80 bytes will be used. The inputs will appear in the registration field in the same order as they are entered.

Example: X" 03420707" MAKE_REG

This will define registration for packet size of 128 bytes.

YES_REGADR **(--)**

Address field will be sent in the Registration packets.

NO_REGADR **(--)**

No address field in the Registration packets. This is the default mode.

The following commands allow the user to set the values for several facilities which will be negotiated in the Call Setup procedure. These facilities are packet- and window size as well as throughput class. Also the user may specify whether or not the facilities reverse charging or fast select will be accepted.
These user specified facilities can be sent within a normally sent Call request packet. If the Call accepted packet following contains values for the facilities these will be accepted, if they are less than or equal to the values sent in the Call request packet.
On the other hand received Call request packets will be accepted if they contain facilities which are specified by the user. Otherwise the Call will be rejected by a Clear request. If the Call received contains values higher than specified, a Call accepted is sent with facility values equal to the specified ones.

FLOW_CONTROL_NEG **(value packet size/ value window size --)**
Set the negotiable packet and window size values.

NO_FLOW_NEG **(--)**
Do not negotiate flow control.

CLASS_NEG **(value --)**
Set the negotiable throughput class value.

NO_CLASS_NEG **(--)**
Do not negotiate throughput class.

YES_REVCH **(--)**
Send/accept reverse charging.

NO_REVCH **(--)**
Do not send/accept reverse charging.

YES_FAST_SEL **(--)**
Send/accept fast select.

NO_FAST_SEL **(--)**
Do not send/accept fast select.

FAST_SEL_RE **(--)**
Send/accept fast select with restriction.

YES_NEG_FAC **(--)**
The user specified facilities will be negotiated.

NO_NEG_FAC **(--)**
The user specified facilities will not be negotiated.

SAVE_FAC **("filename" --)**
Saves the user specified facilities (including values) to disk.

LOAD_FAC **("filename" --)**
To load the user specified facilities saved with the command SAVE_FAC.

This X.25 128 Test Pack gives the user the ability to use basic (mod 8) or extended (mode 128) sequence numbering for the Packet layer.

The following commands are available to choose the sequence numbering for Layer 3 (common to all logical channels) :

L3_MODULO8 **(--)**
This command will set the Layer 3 into modulo 8 mode. The following packets will be sent with a General Format Identifier (GFI) of 0x01.

L3_MODULO128 **(--)**
This command will set the Layer 3 into modulo 128 mode. The following packets will be sent with a General Format Identifier (GFI) of 0x02.

L3_MODULUS **(-- n)**
This commans leaves on the stack the Packet Layer modulus value (8 or 128).

A received packet with an unexpected sequence format will be treated as a GFI error by the emulation. If the monitor is being used with no emulation, it is not necessary to set the modulo with the above FORTH words, as each received packet is decoded according to its GFI bits.

The window sizes for the logical channels can be set in the usual way to any value less than the modulus. The same restrictions as explained for the Link layer apply to the Packet layer, too.

The following commands allow the emulation to simulate a Hunt Group or Call Redirection respectively:

YES_HUNT/ NO_HUNT **(--)**
The facility Hunt Group is/is not subscribed to.

=HUNT_ADDRESS **(adr/busy-flag/n --)**
This command allows the user to set the hunt group addresses.

n=0	indicates the hunt group address
n=1-30	indicate the hunt group member addresses
busy-flag	(#BUSY or #FREE) sets the addresses busy or free (no effect for n=0) .
adr	is a packed string containing the addresses

Usage: `" 1537489" #FREE 1 =HUNT_ADDRESS`
sets the address of hunt group member no. 1 to 1537489 and indicates that this member is able to accept a call.
If a Call request is received with the called address equal to the address of the hunt group (n=0) the next free hunt group member address will be sent in the Call accepted with facility "Called line address modified notification". If no address is available (free) a Clear request is sent. The algorithm to find out the next free address works "round robin".

HUNT_FREE / HUNT_BUSY **(n --)**
sets hunt group member n in free/busy mode.

YES_REDIRECTION / NO_REDIRECTION **(--)**
The facility Call Redirection is/is not subscribed to.

=REDIR_ADDRESS **(adr/busy-flag/n --)**

This command allows the user to set the Call redirection addresses.

n=0	indicates the original address
n=1-7	indicate the alternate addresses
busy-flag	(#FREE or #BUSY) sets the addresses busy or free
adr	is a packed string containing the address

Usage: `" 597348" #BUSY 0 =REDIR_ADDRESS`
sets the original address to 597348 and indicates that this number is busy.

If a Call request is received with the called address equal to the original address to be redirected and this number is busy the next free alternate address will be sent in the Call accepted with facility "Called line address modified notification". If no address is available (free) a Clear request is sent. The algorithm to find out the next free address always starts with address no. 1.

REDIR_FREE /

REDIR_BUSY **(n --)**
sets redirection address no. n in free/busy mode.

3.1.1.2. CHANNEL SPECIFIC SETUP COMMANDS

CH1 **(--)**
Selects CH1 as the current logical channel. Once the current logical channel is selected, the various parameters associated with the selected channels can be set by using the following sequence of commands:

Example: `CH1 SVC ORIG 1 44003300 55005500`

CH1	Selects CH1 logical channel.
SVC	Sets the logical channel to be a switched virtual circuit.
ORIG (or **DEST**)	Sets the logical channel to be used to originate calls or to be a destination. The logical channel number, the called and calling address must be provided after the command ORIG, and the logical channel number must be provided following the command DEST.
1	The logical channel value is entered in decimal as a combined logical channel and group value. Valid numbers are 1 to 4095. If the specified value is greater than 4095, 4095 will be used.
44003300	The called DTE's address field for this logical channel. The maximum address length is 15 decimal digits. (Not required if DEST is used). If length is greater than 15, only the first 15 digits will be used.
55005500	The calling address field for this logical channel. The maximum address length is 15 decimal digits. (Not required if DEST is used). If length is greater than 15, only the first 15 digits will be used.

=CALLED **(string --)**
The command =CALLED will change the called adress field used in call request packets. Use `0 LCNCALLED !` to get a zero length called address field.

Usage: `" address" =CALLED`

=CALLING **(string --)**

The command =CALLING will change the calling address field used in call request packets. Use 0 LCNCALLING ! to get a zero length calling address field.

Usage: " address" =CALLING

LCN **(-- addr)**

The word LCN provides a pointer to the logical channel value for the currently selected logical channel. Thus, any 12 bit value can be stored and will be used as the logical group and channel value.

CH2 / .. / CH3 **(--)**

Functions as the CH1 command for selecting one of the 3 logical channels available.

ECHO_ON **(--)**

All data received in data packets will be transmitted as a data packet on the same logical channel. This command will remain in effect until **ECHO_OFF** is entered.

All T.70 data packets received will be echoed back to the sender. This command will remain in effect until ECHO_OFF is specified.

ECHO_OFF **(--)**

Disable data packet echo. The ECHO_ON and ECHO_OFF commands affect only the selected logical channel (CH1 to CH3). ECHO_OFF is the default setting.

FAST_SEL_OFF **(--)**

The Fast Select facility is not to be used on the specified logical channel. This is the default mode for all logical channels.

Example: CH1 FAST_SELECT_OFF

FAST_SEL_ON **(--)**

The Fast Select facility with no restriction is to be used for the specified logical channel. When FAST_SELECT_ON is active, Clear Request packets will contain calling and called address fields, facility fields and clear user data. The clear confirm packets will contain calling and called address fields and facilities.

Example: CH2 FAST_SELECT_ON

FAST_SEL_RES **(--)**

The Fast Select facility with restriction is to be used for the specified logical channel. When FAST_SELECT_RESTRICTION is active and a Call Request package is received, a Clear Request packet with address, facility and user data fields will be generated.

Example: CH3 FAST_SELECT_RESTRICTION

SET_WINDOW **(value --)**

Set the packet window size per logical channel for sequence numbering used in Layer 3 flow control. Although the user can set the window size to any value, the effective window size achievable is limited by the amount of buffers available, which is in turn determined by the maximum frame length. The default value is set to 2.

Usage: CH1 7 SET_WINDOW

SET_SIZE ({2...2048} --)
Set the data packet size per logical channel. The valid values are from 2 to 1024 bytes. The default value is 128 bytes.
Usage: `CH1 256 SET_SIZE`
Note: The user should be aware of the fact that the frame size is required to be large enough to accommodate the new data packet size.

3.2. SEND PROTOCOL COMMANDS

The SEND PROTOCOL commands use the state machines of the emulation to control the protocol procedure. (See Section 3.3 for a description of the reference emulation state machines). These commands will force protocol state changes and the emulation will expect the correct response. However, only those commands that are appropriate to the current protocol state will be executed; invalid or unexpected commands will be ignored, and no action will be taken. The S: group of send frame and packet commands should be used to generate frames or packets without using the state machines of the emulation. The state machines of the emulation must be activated prior to the use of this group of commands. (Refer to Section 3.3 for commands that turn the state machines ON or OFF).

3.2.1. LAYER 3 SEND COMMANDS

RESTART (--)
A Restart Request packet will be generated and sent. This will clear all the logical channels. The Layer 3 state machines will enter state 3 (WAITING RESTART CONFIRM) after the packet is sent.

CALL (--)
A Call Request packet will be generated and sent on the selected logical channel. The called address and calling address fields are setup with the CHx commands. The Layer 3 state machines will enter LCN state 2 (P2 CALLING).

CALLS (value --)
Send a Call Request packet on several logical channels.

Usage: `x CALLS`

A Call Request packet is sent on each logical channel starting at LCN 1 up to LCN x. All these logical channels will enter LCN state 2 (P2 CALLING) after the execution of the command.

RESET (--)
A Reset Request packet will be generated and sent using the selected logical channel. The Layer 3 state machines will enter LCN state 5 (D2 RESET).

CLEAR (--)
A Clear Request packet will be generated and sent using the selected logical channel. The Layer 3 state machines will enter LCN state 6 (P6 CLEARING).

CLEARS (value --)
Send a Clear Request packet on several logical channels.

Usage: `x CLEARS`

A Clear Request packet is sent on each logical channel starting at LCN 1 up to LCN x. All these logical channels will enter LCN state 6 (P6 CLEARING) after the execution of this command.

INTERRUPT (--)

An Interrupt Request packet is sent on the selected logical channel. This logical channel must already have a call setup.

DATA (--)

A data packet is sent on the selected logical channel using data buffer 0 as the user data. This logical channel must already have a call setup before using this command. Data buffer 0 can support up to 1024 bytes of data.

Note: 1. The data packets are sent only if the packet window is open prior to processing the data packet. If the window is closed, the command is ignored.

2. The term 'selected logical channel' refers to the logical channel which has been activated by the last CHx command or the last received logical channel specific packet (eg. Call Connect packet), received data packet, or received flow control packet.

SENDP (string --)

Send a user defined packet. There are two usages:

(i) ASCII data **Usage:** `" data" SENDP`

(ii) HEX data **Usage:** `X" data" SENDP`

A maximum of 80* bytes can be used by keyboard. For input that is longer that this length, only the first 80 bytes will be used. A maximum of 80 bytes can be entered. For input that is longer than this length, only the first 80 bytes are used. The defined packet will be transmitted in an information frame with the correct frame sequence values.

SENDD (string --)

Send a user defined Data field in a data packet. There are two usages:

(i) ASCII data **Usage:** `" data" SENDD`

(ii) HEX data **Usage:** `X" data" SENDD`

A maximum of 80* bytes of can be used by keyboard. For input that is longer than this length, only the first 80 bytes are used. The data will be transmitted in an data packet with the correct packet sequence values.

Note: When using these commands and more than one Layer2-Link is established, use Link-selection to be on the right link.

example: `3 CES RESTART (Restart is sent on the Link which is selected by the CES=3)`

* For sending Data fields longer than 80 Byte the user has to edit a file which includes the wanted string followed by the appropriate commmand. The maximum string length is 256. After editing thr file has to be executed with the EXEC-Command. The defined packet will have been sent after compiling of the EXEC Command.

3.3. SEND PACKET COMMANDS

The S: send packet commands allow for sending packets at any time without regard to correct protocol procedure. These commands will not update the protocol states, or increment sequence values. They are designed to be used in test scenarios when frames and packets are to be generated out of context.

The send packet commands can be used alone if the Layer 2 state machine is in the information transfer states (states 5 to 8). If the Layer 2 state machine is not in the information transfer states, each send packet command should be followed by the S:INF command to send the I frames which contain the transmitted packets. Each I frame can contain, at most, one packet at a time.

S:RESTARTR (--)

Send a Restart Request packet on logical channel CH0 using values from the following variables:

- LCN – Logical group and channel value
- SENDGFI – GFI modulo value (1 or 2)
- SCAUSE – Cause value
- SDIAG – Diagnostic value

S:RESTARTC (--)

Send a Restart Confirm packet on logical channel CH0 using values from the following variables:

- LCN – Logical group and channel value
- SENDGFI – GFI modulo value (1 or 2)

S:CALLR (--)

Send a Call Request/Incoming Call packet on selected channel using values from the following variables:

- LCN – Logical group and channel value
- SENDDB – D bit value
- SENDGFI – GFI modulo value (1 or 2)
- LCNCALLED – Called address field with the first byte containing the length and with the address bytes in ASCII
- LCNCALLING – Calling address field with the first byte containing the length and with the address bytes in ASCII
- FACILITY-ACTIVE – Indication of type of facility negotiation desired
  - 0 = Facilities not wanted
  - 1 = Negotiate facilities for size, window, an throughput class
  - 2 = Send facilities defined with MAKE_FAC
- CUD-BUFFER – Call user data field buffer. The first byte is the length of the call user data field

S:CALLC (--)

Send a Call Connected/Accepted Call packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDDB	– D bit value
SENDGFI	– GFI modulo value (1 or 2)
RCALLED	– Called address field with the first byte containing the length and with the address bytes in ASCII
RCALLING	– Calling address field with the first byte containing the length and with the address bytes in ASCII
CAFAC-ACTIVE	– Indication of type of facility negotiation desired 0 = NO_CAFAC – do not send facilities 1 = ECHO_CAFAC – echo received facilities from buffer RFAC 2 = USER_CAFAC – Send facilities defined by MAKE_CAFAC

S:CLEARR (--)

Send a Clear Request/Indication packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SCAUSE	– Cause value
SDIAG	– Diagnostic value

When FAST select facility is active, the following are used for Clear Request Packets.

LCNCALLED	– Called address field
LCNCALLING	– Calling address field
FACILITY-ACTIVE	– Indication of type of facility negotiation desired
CUD-BUFFER	– Call user data field buffer. The first byte is the length of the call user data field

S:CLEARC (--)

Send a Clear Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:RESETR (--)

Send a Reset Request/Indication packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SCAUSE	– Cause value
SDIAG	– Diagnostic value

S:RESETC (--)

Send a Reset Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:INTR (--)

Send an Interrupt packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SIUD	– Interrupt user data byte

S:INTC (--)

Send an Interrupt Confirmation packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)

S:DATAP (--)

Send a Data packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDQB	– Q bit value
SENDDB	– D bit value
SENDGFI	– GFI modulo value (1 or 2)
NVS	– Ps value
NVR	– Pr value
SENDM	– More bit value
DATA-BUFFER	– Buffer containing 256 bytes of text
LCNSIZE	– Length of the data field to send

S:RRP (--)

Send a RR packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
NVR	– Pr value

S:RNRP (--)

Send a RNR packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
NVR	– Pr value

S:DIAG (--)

Send a DIAG packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
SDIAG	– Diagnostic value
SDIAG-EXP	– Diagnostic explanation field. The first byte is the length of the field followed by up to 3 bytes for the explanation field to be sent

S:REJP (--)

Send a REJP packet on selected channel using values from the following variables:

LCN	– Logical group and channel value
SENDGFI	– GFI modulo value (1 or 2)
NVR	– Pr value

S:REGISTR (--)

Send a Registration Request packet on selected channel using values from the following variables:

LCN	Logical group and channel value
SENDQB	Q bit value
SENDDB	D bit value
SENDGFI	GFI modulo value (1 or 2)
REGADR-ACTIVE	Indicates, whether Registration address will be sent 0 = Reg. address will not be sent 1 = Reg. address will be sent

(For REGADR-ACTIVE = 1)

LCNCALLED	Called address field with the first byte containing the length and with the address bytes in ASCII
LCNCALLING	Calling address field with the first byte containing the length and with the address bytes in ASCII
REGIST-ACTIVE	Indicates, whether Registration field will be sent 0 = Reg. field not wanted 2 = Send Reg. field defined by MAKE_REG

S:REGISTC (--)

Send a Registration Request packet on selected channel using values from the following variables:

LCN	Logical group and channel value
SENDQB	Q bit value
SENDDB	D bit value
SENDGFI	GFI modulo value (1 or 2)
REGADR-ACTIVE	Indicates, whether Registration address will be sent 0 = Reg. address will not be sent 1 = Reg. address will be sent

(For REGADR-ACTIVE = 1)

LCNCALLED	Called address field with the first byte containing the length and with the address bytes in ASCII
LCNCALLING	Calling address field with the first byte containing the length and with the address bytes in ASCII
REGIST-ACTIVE	Indicates, whether Registration field will be sent 0 = Reg. field not wanted 2 = Send Reg. field defined by MAKE_REG
SCAUSE	Cause value
SDIAG	Diagnostic value

4. TEST MANAGER

4.1. EVENT RECOGNITION WORDS

4.1.1. ?RX_PACKET

The **?RX_PACKET** word is for use within the event field to check whether one of a list of Layer 3 events has been received.

Format: `EVENT1 ... EVENTN n ?RX_PACKET`
where n is the total number of events to be checked, and the events can be any one of the following:

EVENT	DESCRIPTION
R*DATAP	Received Data packet
R*RRP	Received RR packet
R*REJP	Received RNR packet
R*CALLREQ	Received Call Request packet
R*CALLCON	Received Call Connected packet
R*CLEARREQ	Received Clear Request packet
R*CLEARCONF	Received Clear Confirmation packet
R*INTREQ	Received Interrupt Request packet
R*INTCONF	Received Interrupt Confirmation packet
R*RESETREQ	Received Reset Request packet
R*RESETCONF	Received Reset Confirmation packet
R*RESTARTREQ	Received Restart Request packet
R*RESTARTCONF	Received Restart Confirmation packet
R*INVPKT	Received Invalid packet
R*DIAGNOSTIC	Received Diagnostic packet

This word is intended to be used in the case where Layer 2 is running automatically and the Layer 3 state machine is not to be executed. This allows the test scenario to have total control of packet responses.

Example:

```
1 STATE{  R*CALLREQ  1  ?RX_PACKET          ( check for Call Request packet )
               ACTION{  ." Call Request received"  CR
               }ACTION
}STATE
```

4.1.2. ?RX

The **?RX** word is used within the event field to check whether one of a list of packets has been received.

Format: `EVENT1 . . . EVENTN n ?RX`
where N is the total number of events, and the event may be any one of the packet events listed in the ?RX_PACKET list.
This word is intended to be used in the case where layer 2 is running automatically and layer 3 State-machine is to be executed too.
That means the testscenario can make use of the L3 Emulator as long as necessary, before getting the control of packet responses.

4.1.3. LCN-MULTIPLEXER

=CN_LCN (lcn-value --)
select a logical channel number for multiplexing. The current connection only listens to packets with this LCN and to packets with LCN=0!

?CN_LCN (-- lcn-value)
returns the selected logical channel

=CN_CH (ch-value --)
select a channel structure this connection

?CN_CH (-- ch-value)
returns the selected channel structure

Example:

```
0 STATE_INIT{
      L3_MUX_ON    L3_OFF
      #UNUSED      =CN_LCN
      1            =CN_CH            ( sets the channel structure
                                       to 1. Ie., if the incoming
                                       frame pertains to this
                                       connection, the current
                                       channel structure is updated
                                       to the selected one (CH1))
}STATE_INIT

0 STATE{
      R*RESTARTREQ 1 ?RX_PACKET
      ACTION{
                   MDL_CURRENT =CN_CEI
                   S:RESTARTC
      }ACTION

      R*CALLREQ    1 ?RX_PACKET
      ACTION{
                   M-LCN@ =CN_LCN    (sets the LCN for this
                                      connection with received LCN.
                                      Ie., only packets with the
                                      selected channel number are
                                      routed to this connection)
                   S:CALLC
      }ACTION
}STATE
```

4.2. LIST OF COMMUNICATION VARIABLES

The X.25 program utilizes variables to hold the decoded information from received frames. The following list gives a description of the variables that can be utilitzed the Test Manager action field to analyze the protocol procedure.

PKT-LENGTH	(-- addr)	Length of the received packet.
PKT-POINTER	(-- addr)	Pointer to the first byte of the received packet.
M-GFI	(-- addr)	GFI value (1=Modulo 8, 2=Modulo 128).
M-Q	(-- addr)	Q bit.
M-D	(-- addr)	D bit.
M-LCN	(-- addr)	Pointer of logical group and channel value. (12 bit)
M-REC-PKT-ID	(-- addr)	Packet type identifier byte.
M-RCALLED	(-- addr)	Array of 16 byte which contains called address, first byte is length. The address is stored as ASCII characters.
M-RCALLING	(-- addr)	Array of 16 byte which contains calling address, first byte is length. The address is stored as ASCII characters.
M-RFAC	(-- addr)	Array of 128 byte which contains received facility field. First byte is length.
M-RCUD	(-- addr)	Array of 256 byte which contains received Call or Clear user data field. First byte is length.
M-RCAUSE	(-- addr)	Cause byte.
M-RDIAG	(-- addr)	Diagnostic byte.
M-DIAGNOSTICF	(-- addr)	Pointer of Diagnostic field from diagnostic packet.
M-MORE	(-- addr)	More bit from data packet.
M-PS	(-- addr)	PS value.
M-PR	(-- addr)	PR value.
DATA-LENGTH	(-- addr)	Length of Data field in data packets.
DATA-POINTER	(-- addr)	Pointer to first byte of data field in data packet.

LAYER 3

DATA-LENGTH	(-- addr)	Length of the received data packet field
DATA-POINTER	(-- addr)	Address of the received data packet field
PR	(-- addr)	Received PR sequence
PS	(-- addr)	Received PS sequence
PKT-LENGTH	(-- addr)	Length of received packet layer
PKT-POINTER	(-- addr)	Address of received packer layer
RCALLED	(-- addr)	16 byte received calling address (count prefixed)
RCALLING	(-- addr)	16 byte received called address (count prefixed)
RCAUSE	(-- addr)	Received cause field
RCUD	(-- addr)	128 byte received call user data field
RDIAG	(-- addr)	Received diagnostic field
REC_PKT_ID	(-- addr)	ID byte received call user data field
RECD	(-- addr)	Received D bit
RECM	(-- addr)	Received M bit
RECPKTMOD	(-- addr)	Received packet Modulo ID from GFI
RECQ	(-- addr)	Received Q bit
RFAC	(-- addr)	64 byte received facility field
RIUD	(-- addr)	Received Interrupt User Data byte from Interrupt Request packets
RLCN	(-- addr)	Received LCN value
SCAUSE	(-- addr)	Cause field for transmitted RESET, RESTART, CLEAR packets
SDIAG	(-- addr)	Diagnostic field for transmitted RESET, RESTART, CLEAR packets
SDIAG-EXP	(-- addr)	Diagnostic explanation field. The first byte is the length of the field followed by up to 3 bytes for the explanation field to be sent.
SENDM	(-- addr)	Value of next M bit to be sent
SIUD	(-- addr)	Interrupt User Data buffer used for transmitting Interrupt Request packets
STATE_L3	(-- addr)	Layer 3 state variable
T20_VALUE	(-- addr)	T20 RESTART timer
T21_VALUE	(-- addr)	T21 CALL timer
T22_VALUE	(-- addr)	T22 RESET timer
T23_VALUE	(-- addr)	T23 CLEAR timer

SENDQB	(-- addr)	SENDQB is a variable which contains the value of the Q bit used for sending packets. The valid value is 0 or 1.
SENDDB	(-- addr)	SENDDB is a variable which contains the value of the D bit used for sending packets. The valid value is 0 or 1.
SENDGFI	(-- addr)	SENDGFI is a variable which contains the value of the GFI modulo indicator used for sending packets. The valid value is 0X10 or 0X20.

LAYER 3 - LOGICAL CHANNEL VARIABLES

The logical channel variables access the values pertaining to the current logical channel selection (CH1 to 3).

LCN	(-- addr)	Selected logical channel number.
LCNSTATE	(-- addr)	State for Reference emulation.
LCNWINDOW	(-- addr)	Window size for selected logical channel.
LCNSIZE	(-- addr)	Size of data field used with DATA command.
NVS	(-- addr)	LCN Send Sequence variable. This is the next PS that will be sent.
NVR	(-- addr)	LCN Receive Sequence variable. This is the next PS expected to be received.
NSU	(-- addr)	LCN sequence number (PS) of the lowest unacknowledged Data Packet that was sent.
LCNCALLING	(-- addr)	LCN calling address field
LCNCALLED	(-- addr)	LCN called address field.

COMMANDS

=CN_LCN	**(lcn-value --)** sets the logical channel number for currently selected connection
?CN_LCN	**(-- lcn-value)** returns the logical channel number for currently selected connection
=CN_CH	**(ch-value --)** sets the logical channel for currently selected connection
?CN_CH	**(-- ch-value)** returns the logical channel for currently selected connection

INDEX

Symbols

?CN_CH, 4-2, 4-5
?CN_LCN, 4-2, 4-5
?RX, 4-1
?RX_PACKET, 4-1
=CALLED, 3-6
=CALLING, 3-7
=CN_CH, 4-2, 4-5
=CN_LCN, 4-2, 4-5
=HUNT_ADDRESS, 3-5
=REDIR_ADDRESS, 3-6

C

CALL, 3-8
CALLS, 3-8
CCITT-80, 3-1
CCITT-84, 3-1
CH1, 3-6
CH2/../CH3, 3-7
CLASS, 3-1
CLASS_NEG, 3-4
CLEAR, 3-8
CLEARS, 3-8
COMMAND, 2-3

D

DATA, 3-9
DATA-LENGTH, 4-3, 4-4
DATA-POINTER, 4-3, 4-4
DEST, 3-6

E

ECHO_CAFAC, 3-2
ECHO_OFF, 3-7
ECHO_ON, 3-7

F

FAST_SEL_OFF, 3-7
FAST_SEL_ON, 3-7
FAST_SEL_RE, 3-4
FAST_SEL_RES, 3-7
FLOW_CONTROL_NEG, 3-4
FREEZE, 2-3

H

HALT, 2-3
HELP, 2-3
HUNT_BUSY, 3-5
HUNT_FREE, 3-5

I

INTERRUPT, 3-9

L

L3_MODULO128, 3-5
L3_MODULO8, 3-5
L3_MODULUS, 3-5
L3_OFF, 3-14
L3_ON, 3-14
LCN, 3-7, 4-5
LCNCALLED, 4-5
LCNCALLING, 4-5
LCNSIZE, 4-5
LCNSTATE, 4-5
LCNWINDOW, 4-5
LOAD_FAC, 3-4

M

M-D, 4-3
M-DIAGNOSTICF, 4-3
M-GFI, 4-3
M-LCN, 4-3
M-MORE, 4-3
M-PR, 4-3
M-PS, 4-3
M-Q, 4-3
M-RCALLED, 4-3
M-RCALLING, 4-3
M-RCAUSE, 4-3

M-RCUD, 4-3
M-RDIAG, 4-3
M-REC-PKT-ID, 4-3
M-RFAC, 4-3
MAKE_CAFAC, 3-2
MAKE_CUD, 3-3
MAKE_FAC, 3-2
MAKE_REG, 3-3
MENU, 2-3
MON_KEYS, 2-3
MONITOR, 2-3

N

NEW_L3_STATE, 3-14
NEW_LCN_STATE, 3-14
NO_CA, 3-2
NO_CAFAC, 3-2
NO_CLASS_NEG, 3-4
NO_FAC, 3-1
NO_FAST_SEL, 3-4
NO_FLOW_NEG, 3-4
NO_HUNT, 3-5
NO_NEG_FAC, 3-4
NO_REDIRECTION, 3-5
NO_REG, 3-3
NO_REGADR, 3-3
NO_REVCH, 3-4
NSU, 4-5
NVR, 4-5
NVS, 4-5

O

ORIG, 3-6

P

PKT-LENGTH, 4-3, 4-4
PKT-POINTER, 4-3, 4-4
PR, 4-4
PS, 4-4

Q

QUIT_COM, 2-3
QUIT_HELP, 2-3
QUIT_MENU, 2-3

R

RCALLED, 4-4
RCALLING, 4-4
RCAUSE, 4-4
RCUD, 4-4
RDIAG, 4-4
REC_PKT_ID, 4-4
RECD, 4-4
RECM, 4-4
RECPKTMOD, 4-4
RECQ, 4-4
REDIR_BUSY, 3-6
REDIR_FREE, 3-6
RESET, 3-8
RESTART, 3-8
RFAC, 4-4
RIUD, 4-4
RLCN, 4-4

S

S:CALLC, 3-11
S:CALLR, 3-10
S:CLEAR, 3-11
S:CLEARC, 3-11
S:DATAP, 3-12
S:DIAG, 3-12
S:INTC, 3-12
S:INTR, 3-12
S:REGISTC, 3-13
S:REGISTR, 3-13
S:REJP, 3-13
S:RESETC, 3-11
S:RESETR, 3-11
S:RESTARTC, 3-10
S:RESTARTR, 3-10
S:RNRP, 3-12

S:RRP, 3-12
SAVE_FAC, 3-4
SCAUSE, 4-4
SDIAG, 4-4
SDIAG-EXP, 4-4
SENDD, 3-9
SENDDB, 4-5
SENDGFI, 4-5
SENDM, 4-4
SENDP, 3-9
SENDQB, 4-5
SET_SIZE, 3-8
SET_WINDOW, 3-7
SIUD, 4-4
STATE_L3, 4-4
SVC, 3-6

T

T10-VALUE, 3-14
T11-VALUE, 3-14
T12-VALUE, 3-14
T13-VALUE, 3-14
T20-VALUE, 3-14
T20_VALUE, 4-4
T21-VALUE, 3-14
T21_VALUE, 4-4
T22-VALUE, 3-14
T22_VALUE, 4-4
T23-VALUE, 3-14
T23_VALUE, 4-4
TM_KEYS, 2-3

U

USER_CAFAC, 3-2
USER_FAC, 3-1
USER_REG, 3-3

Y

YES_CA, 3-2
YES_FAC, 3-1
YES_FAST_SEL, 3-4
YES_HUNT, 3-5
YES_NEG_FAC, 3-4
YES_REDIRECTION, 3-5
YES_REGADR, 3-3
YES_REVCH, 3-4